Master's Thesis in the International Master of Science in Computational Mechanics

By

KHUONG ANH DUNG

Title

IMPOSING CONTACT CONDITION WITH THE EXTENDED FINITE ELEMENT METHOD (X-FEM) ON NON-MATCHING GRIDS

Supervisor

Prof. **NICOLAS MOËS**

The International Center for Numerical Methods in Engineering (CIMNE) &
Research Institute of Civil Engineering and Mechanics, Institut GeM - UMR CNRS 6183
Ecole Centrale de Nantes

May 2009

ABSTRACT

Imposing contact condition with X-FEM on non-matching grids

Khuong Anh Dung

The extended finite element method (X-FEM) has been developed to minimize requirements on the mesh design in a problem with a displacement discontinuity. In this report, the imposition of kinematic conditions along interfaces modeled with X-FEM is studied. In order to model the kinematic condition, our aim is to build a Lagrange multiplier space. The stability of the formulation is ensured by a LLB (inf-sup) fulfilling algorithm. By introducing a Lagrangian multiplier space, this method realizes an optimal control for the interface problem. A non-matching finite element grid on the interface is considered and an optimal energy-norm error estimate in the finite time is obtained.

In addition, a model of the contact condition by X-FEM was proposed and applied to test analysis. The computation has been made to ensure the accuracy. Some results of numerical analysis by X-FEM are presented. Numerical examples show that the convergence rates are preserved and the inf-sup conditions are passed. The results obtained for these test cases are logical.

ACKNOWLEDGEMENTS

The author would like to extend my sincerest thanks and regards to all those who helped me to reach this level. I would like to thank the myriad persons who provided support and encouragement throughout the pursuit of this project.

Firstly, I thank the European Commission for a student scholarship and stipend during the master programme. I greatly appreciate the international master of science in Computational Mechanics, founded by an international consortium of four leading European Universities: Universitat Politècnica de Catalunya (UPC), Swansea University, Universität Stuttgart and Ecole Centrale de Nantes (ECN) in cooperation with the International Center for Numerical Methods in Engineering (CIMNE).

I would like to thank my committee for taking the time to read my thesis and listen to my defense. This project has been carried out with Prof. **Nicolas MOËS** as supervisor who has kindly spent so much time on my notes and answered so many questions patiently. He has helped me greatly and been a source of knowledge. I am grateful to him for his endless support and for generously sharing his time and knowledge. I will always remain grateful to him.

I would like to thank Dr. **Nicolas CHEVAUGEON** for his comments and suggestions of practical works and programming as well as excellent instruction and a general helping hand.

I would also like to thank Prof. **Pedro Diez,** CIMNE, for his co-operation and involvement. He has played a major role in helping and managing the master course.

I would like to express my special thanks to **Lelia Zielonka** and **Maria**, CIMNE and **Anne-Laure FREMONDIERE**, ECN, the graduate program coordinator, deserves special mention. Their relentless work to ensure that graduate students have what they need is an asset to the department. They are always ready to help with a smile. I thank to **Eric Manceau**, **Gilles Marckmann** for providing "IT support".

I am indebted to my many classmates for providing a stimulating and fun environment in which to learn and grow.

Thanks also to my parents, who provided the item of greatest worth - opportunity. Thank you for standing by me through the many trials and decisions of my educational career.

For all the above named persons and other nameless people, who positively contributed in my life I express my thanks and appreciation.

Finally, it should be noted that the thesis could never have been completed without the sense of high quality and professionalism of the faculty members of UPC and ECN. They all gave me the honor of attaining the Master degree.

KHUONG Anh-Dung

Contents

List of figures

List of tables

Chapter 1
Introduction

In spite of the important effort dedicated to X-FEM in the last decade for a general presentation, there are still many aspects that require further research. An efficient imposition of essential boundary conditions in X-FEM is still an open issue. Finite element shape function satisfies “Kronecker delta” property $N_i(\mathbf{x}_j) = \delta_{ij}$ which makes imposition of essential boundary conditions straightforward. In recent years, many specific techniques for the implementation of essential boundary conditions in X-FEM have been developed. In certain X-FEM cases, imposition of essential boundary conditions is tricky. There are certain difficulties in imposing essential boundary conditions. Imposing essential boundary conditions is one of a key issue in X-FEM. Probably the simplest way to impose essential boundary conditions is by boundary collocations. Another opportunity is based on a modification of the weak form, such the penalty method [17], Lagrange multipliers [16] or Nitsche’s method [25]. These methods consider a modified weak form and they allow the use of trial functions that do not vanish at the essential boundary. In fact, the penalty method and Nitsche's method require only the choice of one scalar parameter. In the penalty method, large values of this parameter must be used in order to impose the essential boundary condition in a proper manner. In practice, that leads to ill-conditioned systems of equations, reducing the applicability of this method. On the contrary, Nitsche's method does not suffer of ill-conditioning. However, the implementation of Nitsche's method is not as trivial as for the Lagrange multiplier method or the penalty method, in the sense that the modification of the weak form is different for each particular problem.

In the X-FEM context, the Lagrange multiplier method is one of the most widely used because of its straightforward implementation in all kind of problems. This method introduces a new unknown function, the Lagrange multiplier. The interpolation space for the Lagrange multiplier must be carefully selected: it has to be rich enough in order to obtain an acceptable solution. The aim of this project is to review and compare some of the most powerful techniques for the imposition of essential boundary conditions X-FEM. In this report we propose a Lagrange multiplier space to model the kinematic condition. It can be applied to the imposition of essential boundary conditions.

Special attention is paid to the choice of Lagrange multipliers with X-FEM proposed in [29] and to methods based on a modification of the weak form. Section 2 recalls basic concepts on the X-FEM. Section 3 is devoted to review and compare three techniques based on a modification of the weak form: the Lagrange multiplier method, the penalty method, and Nitsche's method. Also, LBB condition has been discussed. Finally, in Section 4, two numerical examples corroborate the conclusions. In this thesis, various applications of the X-FEM were developed and applied to model the kinematic condition along the interface. We prove energy type in 2 space dimensions, and present some numerical examples.

Chapter 2

Overview of the partition of unity, level sets and the eXtended Finite Element Method (X-FEM)

Modelling discontinuities and singularities in the approximating space has been remained a challenge in the world of computational mechanics. The standard Finite Element Method when employed for modelling problems containing discontinuities such as cracks, material interfaces, high gradients etc. , poses a problem in a sense that the FEM mesh is required to conform with the geometry of the discontinuity. Additionally in order to capture high gradients in the approximating field, significant refinement in the vicinity of the discontinuity is required. Above all modelling discontinuities evolving in time with FEM is burdensome due to the need to update the mesh to match the geometry of the discontinuity. This also increases the computational cost. To alleviate the shortcomings associated with meshing of cracks using FEM, Partition of Unity Methods (PUM) were developed. X-FEM is a partition of unity based enriched/extended finite element method.

The work is based on developments of the eXtended Finite Element Method, which has been successfully applied to static problems exhibiting discontinuities or heterogeneities such as cracks, holes or material interfaces. The governing idea of this method is to enrich the classical FEM approximation based on the Partition of Unity technique with specific functions representing surfaces of discontinuities or heterogeneities. Level sets are used to locate the physical surfaces on the mesh. Their sign indicate the side on which a point is located. Level sets use node-valued functions and are interpolated with the basic functions of the finite element. This description allows to release the underlying mesh from the description of surfaces of discontinuity or external boundaries.

2.1 Partition of unity method

Melenk and Babuska [24] did show that the traditional finite element approximation could be enriched so as to represent a specified function on a given domain. Their point of view can be summarized as follows [28]. Let first us recall that the finite element approximation is written on an element as

$$u(x)\big|_{\Omega_e} = \sum_{i \in N_n} \sum_{\alpha} a_i^{\alpha} \phi_i^{\alpha}(x) \tag{2.1}$$

As the degrees of freedom (dof.) defined at a node have the same value for all the elements connected to it. The approximations on each element can be “assembled” to give a valid approximation in any point x of the domain

$$u(x) = \sum_{i \in N_n} \sum_{\alpha} a_i^{\alpha} \phi_i^{\alpha}(x) \tag{2.2}$$

where $N_n(x)$ is the set of nodes belonging to the elements containing point x. The domain of influence (support) of the approximation function ϕ_i^α is the set of elements connected to node i. The set $N_n(x)$ is thus also the set of the nodes whose support covers point x.

It is thus possible to enrich the finite element approximation by the same techniques as those used in meshless methods. Here is the enriched approximation which makes it possible to represent function $F(x)e_x$ on domain Ω_F

$$u(x) = \sum_{i \in N_n(x)} \sum_{\alpha} a_i^\alpha \phi_i^\alpha + \sum_{i \in N_n(x) \cap N_F} \sum_{\alpha} b_i^\alpha \phi_i^\alpha(x) F(x) \tag{2.3}$$

where N_F is the set of nodes whose support has an intersection with domain Ω_F. The proof is obtained by setting to zero coefficients a_i^α and by taking into account the fact that the finite element shape functions are able to represent all rigid modes and thus the e_x mode. We will see the concrete use of the partition of unity for the modelling of discontinuities in the following part.

2.2 Level Sets

The Osher-Sethian level set method [51] tracks the motion of an interface by embedding the interface as the zero level set of the signed distance function. The motion of the interface is matched with the zero level set of the level set function, and the resulting initial value partial differential equation for the evolution of the level set function resembles a Hamilton-Jacobi equation. In this setting, curvatures and normals may be easily evaluated, topological changes occur in a natural manner, and the technique extends trivially to three dimensions. This equation is solved using entropy-satisfying schemes borrowed from the numerical solution of hyperbolic conservation laws which produce the correct viscosity solution.

Level set method is designed for problems in which the speed function can be positive in some places are negative in others, so that the front can move forwards in some places and backwards in others.

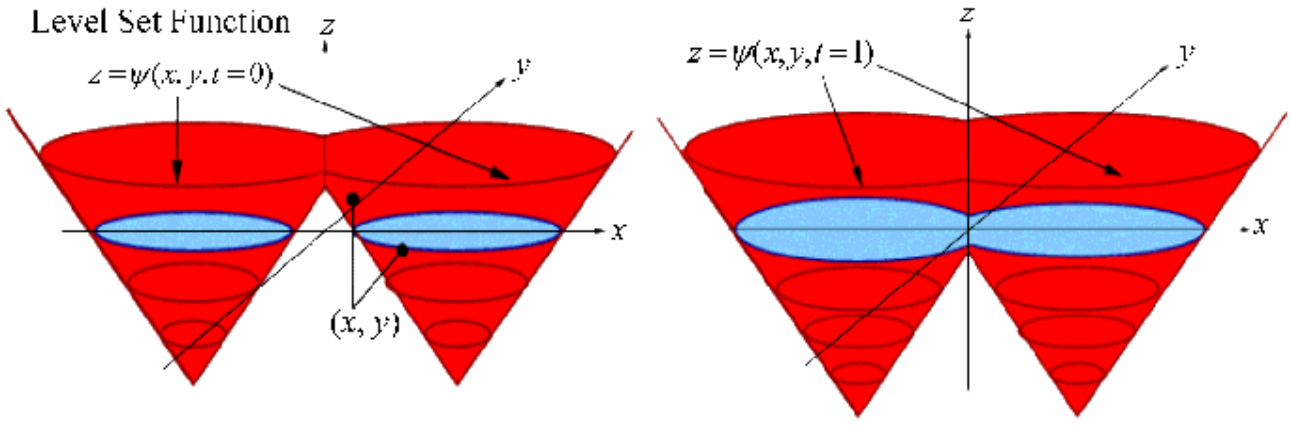


Figure 2.1. *Example of 2D curve propagation with the level set method (taken from [9]).*

In the level set method, the boundary is represented implicitly as an iso-surface of a function, a so-called level set function, of a higher dimension. The resulting evolution is then mapped into an evolution of the level set function. The advantages of this method are that the method handles changes of topology naturally without special rules for collision detection, can be easily adapted to any number of dimensions, and can treat the formation of corners and cusps in the boundary correctly through the use of methods borrowed from hyperbolic conservation laws.

This method encapsulates the entire evolution of the surface with a single time-independent function. While it is typically only used for monotonic speed functions, i.e. fronts which move only in one direction, recent work shows that this restriction can be overcome in some instances.

The level set of a differentiable function: $f:\mathbb{R}^n \to \mathbb{R}$ corresponding to a real value c is the set of points

$$\left\{\left(x_1,...,x_n\right)\in\mathbb{R}^n : f\left(x_1,...,x_n\right)=c\right\} \tag{2.4}$$

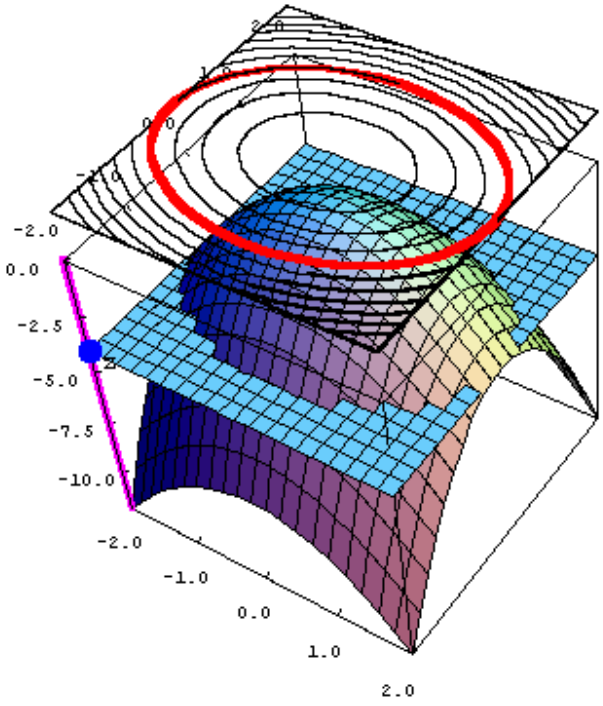


Figure 2.2. *Example of a level set representation of a circle (bold curve). Red curve is zero level set (taken from [20]).*

Within the framework of X-FEM another development is the use of Level set functions for representing discontinuities. Where the discontinuity is represented as a zero Level set function. The surfaces are located and evolved by the level set technique.

2.3 The eXtended Finite Element Method (X-FEM)

2.3.1 Basic

Based on an idea of Lancaster and Salkauskas [40] and probably motivated by the purpose to model arbitrary crack propagation without computational expensive remeshing, the group of Prof. Ted Belytschko developed the X-FEM. Since its introduction in 1999 (Moës, Dolbow, and Belytschko 1999 [30]), the extended finite element method (X-FEM) has been successfully developed and applied in science and engineering computations. The extended finite element method is currently widely used in research and starts to appear in industry. In general X-FEM can be used to model cracks, contact and interface problems, simulation of inclusions and holes, moving discontinuities, bi-phase flow problems, fluid structure interaction, etc. This method basically eliminates the need to mesh physical surfaces in finite element computations.

The finite element method is used as base in X-FEM, and hence large body of the finite element method can be readily exploited. The eXtended Finite Element Method exploits the partition of unity property of finite elements, which allows local enrichment functions to be easily incorporated in the FEM approximation. As the name implies, the standard FEM approximation space is extended or enriched using an appropriate enrichment function, which can best describe the field. Additional functions which might contain any prior knowledge/information about the solution can be incorporated into the finite element space using partition of unity and the resulting space is capable of capturing the local features of interest. For crack modelling a discontinuous jump function and asymptotic near tip displacement fields are added to the displacement-based finite element approximation. The whole beauty of X-FEM lies in subdividing the problem into two parts: part A) generating mesh without cracks/inclusions, etc.; part B) enriching the FEM approximation with additional/enrichment functions that models the discontinuities. This avoids the need for remeshing or explicit geometric modelling of the discontinuity.

The enriched finite element approximation is written:

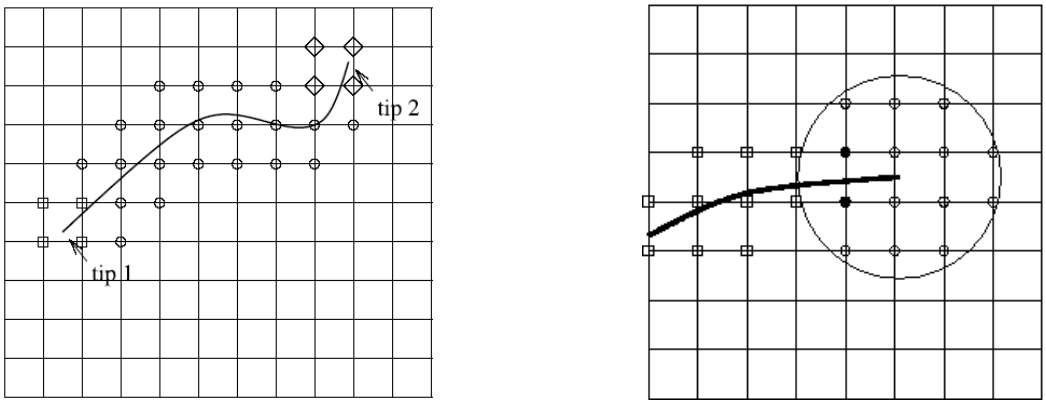
$$u_i(\boldsymbol{x}) = \sum_{J\in S} u_{Ji}\,\Phi_J(\boldsymbol{x}) + \sum_{J\in S_c} a_{Ji}\,H(f(\boldsymbol{x}))\,\Phi_J(\boldsymbol{x}) + \sum_{J\in S_{ct}} \Phi_J(\boldsymbol{x})\,b_{Jij}B_j(\boldsymbol{x}) \tag{2.5}$$

where

$$H(f(\boldsymbol{x})) = \begin{Bmatrix} +1 & f(\boldsymbol{x}) > 0 \\ -1 & f(\boldsymbol{x}) < 0 \end{Bmatrix} \text{ and } B_j(\boldsymbol{x}) = \left(r^n \sin\frac{\theta}{2} \right) \tag{2.6}$$

In the classical FE approximation, to represent the discontinuity at the interface, the interface needs to be aligned with the mesh.

$$\begin{aligned} T^h(x) &= \sum_I \phi_I(x) T_I \\ \nabla T^h(x) &= \sum_I \nabla\phi_I(x) T_I \end{aligned} \tag{2.7}$$



(a) (b)

Figure 2.3. *Node selection for the enrichment in X-FEM: (a) the circled nodes are enriched by the jump function whereas the squared nodes are enriched by the crack tip functions [30]; (b) the squared nodes are enriched by the jump function whereas the circled nodes are enriched by the crack tip functions.*

In the X-FEM approximation, the interface can be represented by evolving g(x)

$$T^h(x) = \sum_I \phi_I(x) T_I + \sum_J \phi_J(x) \bullet g(x) a_J$$
$$\nabla T^h(x) = \sum_I \nabla \phi_I(x) T_I + \sum_J \left(\nabla \phi_J(x) \bullet g(x) + \phi_J \bullet \nabla g(x) \right) a_J \quad (2.8)$$

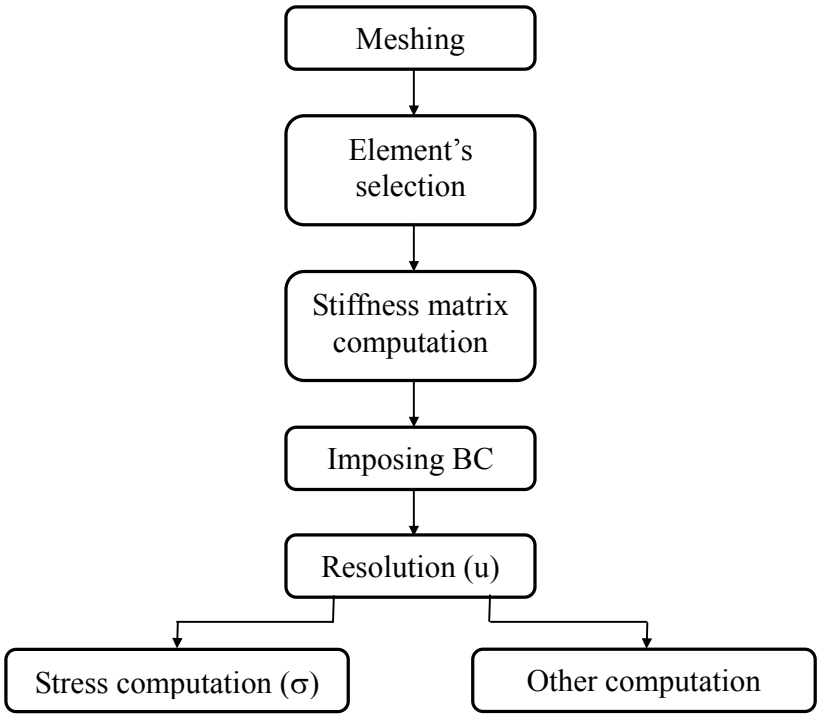


Figure 2.4. *Structure of the X-FEM code*.

➢ **X-FEM on element level**

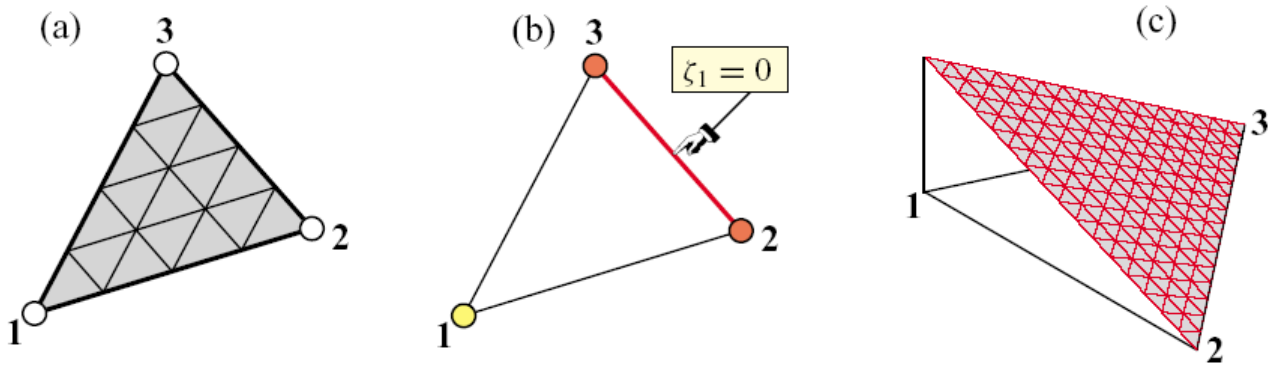


Figure 2.5. *The three-node linear triangle: (a) element geometry; (b) equation of side opposite corner 1; (c) perspective view of the shape function* $N_1 = \zeta_1$.

➢ **The 3-node triangular element**

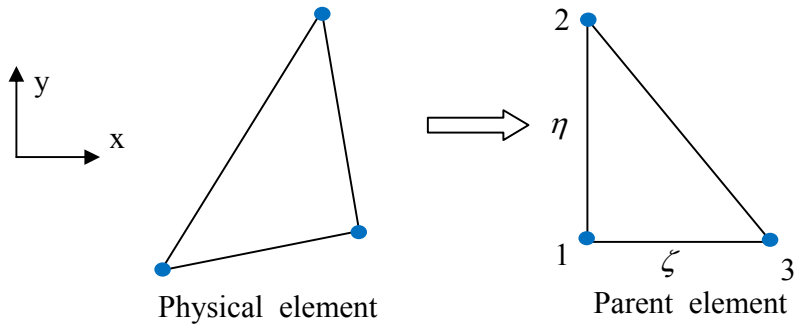
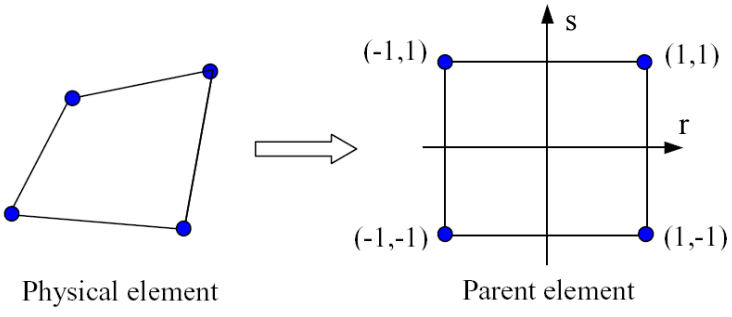


Figure 2.6. *Physical and parent 3-node elements.*

➢ **4-node quadrilateral (linear shape functions)**



$$N_1(r,s) = \frac{1}{4}(1-r)(1-s)$$
$$N_2(r,s) = \frac{1}{4}(1+r)(1-s)$$
$$N_3(r,s) = \frac{1}{4}(1+r)(1+s)$$
$$N_4(r,s) = \frac{1}{4}(1-r)(1+s) \qquad (2.9)$$

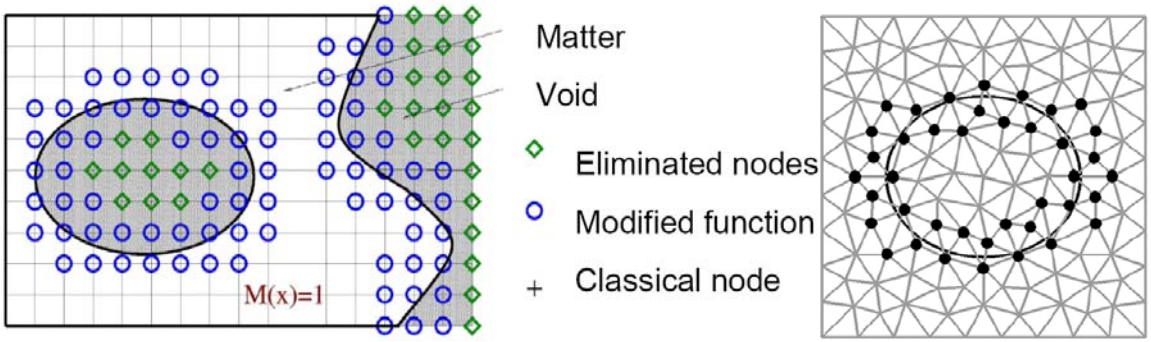
Figure 2.7. *Physical and parent 4-node elements.*

2.3.2 Modelling holes and material interfaces with the X-FEM

We show here how to model holes which are not defined by the mesh. The idea is to define an enrichment function $V(x)$ which is zero in the holes and one in the body

$$V(x) = \begin{cases} 1 & if \ x \in \Omega \\ 0 & if \ x \notin \Omega \end{cases} \tag{2.10}$$

where Ω is the domain occupied by the body. If the support of a nodal shape function intersects a hole, the nodal shape function is multiplied by the $V(x)$ function so that the support size is reduced to its material fraction. Also, the nodal degrees of freedom for which the supports are totally in the void are eliminated (or set to zero depending on the implementation). Figure 2.8 (a) illustrates the enrichment strategy for a body containing hole and void.



(a) (b)

Figure 2.8. *Example of (a) enrichment for a discontinuity: eliminated nodes (dof. set to zero), modified function (shape function multiplied by V(x)) [6]; (b) non-conforming mesh to model an inclusion [41].*

In the X-FEM context, its main characteristic is to separate the problem into two parts [41]:

- The first part corresponds to the discretization of whole domain, which does not include some or all the surfaces related to discontinuities or boundary conditions. The approximation of the displacement field is consequently the classical approximation used in FEM, and mesh generation is generally straightforward.

- The second part (for material interfaces) consists in supplementary shape functions added to some nodes of the previous approximation. The goal of these additional functions is to enrich the basic approximation of the existing displacement field with less regular functions able to model jumps on the surfaces which are not meshed. In Figure 2.8 (b), black circles indicate these nodes. The determination of these nodes can be made with the help of a level set representation of the surface [35]. When nodes have been selected, specific enrichment functions are then associated to these and offer for instance the possibility to take the deformation discontinuity at a material interface into account.

2.3.3 Comparison of the classical FEM and X-FEM in 1D model problem

Problem with homogeneous Dirichlet boundary conditions and initial conditions.

$$\begin{cases} u - u_{,xx} = f \\ u(a) = u_0 \quad IC \\ u(b) = u_1 \quad BC \end{cases} \tag{2.11}$$

where

$$f = f' - 32\delta(x_f) \qquad f' = \begin{cases} 16x^4 - 192x^2 & x < x_f \\ (x-1)^4 - 12(x-1)^2 & x \geq x_f \end{cases} \tag{2.12}$$

Analytical solution [14]: (note that the derivative of u is discontinuous at point x_f)

$$u = \begin{cases} 16x^4 & x < x_f \\ (x-1)^4 & x \geq x_f \end{cases} \tag{2.13}$$

Numerical results of 1D problem

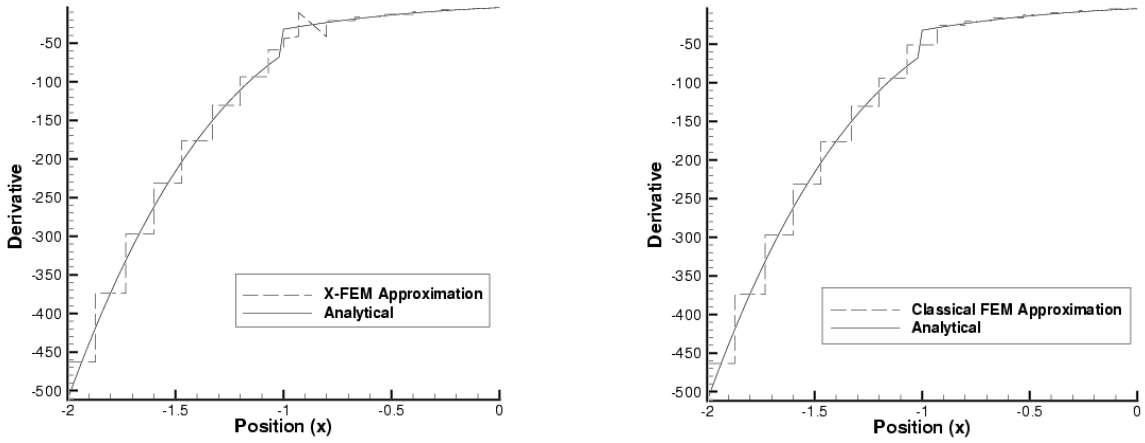
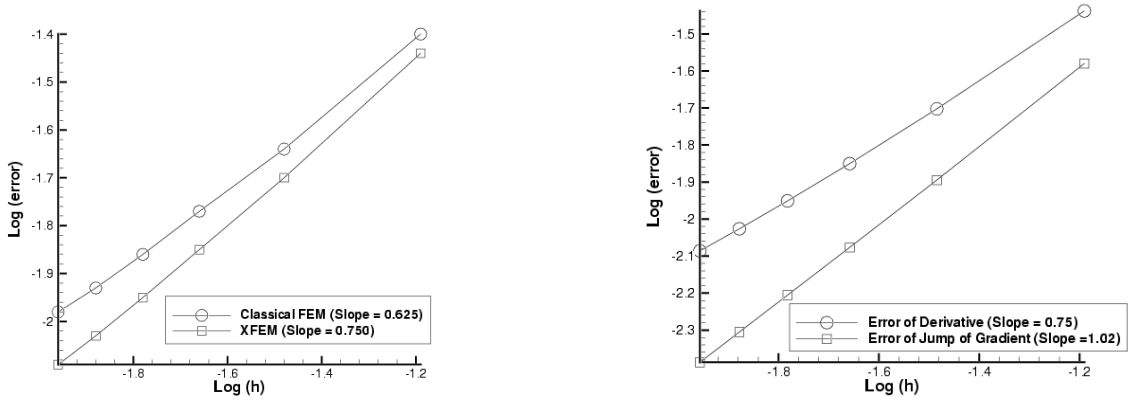


Figure 2.9. *Comparison of derivative using classical FEM and X-FEM.*



(a) (b)

Figure 2.10. *(a) Comparison of the rate of convergence using classical FEM and X-FEM - the phase boundary is always located between nodes; (b) Rate of convergence using Lagrange multipliers.*

The X-FEM permits to have a better accuracy than the classical FEM with an equivalent time of computation.

Chapter 3
Stability study

3.1 Dirichlet boundary conditions with X-FEM

Trace of the inner field on the Dirichlet boundary is very rich when the boundary is not matched. If the strong imposition of Dirichlet (naive approach) is implied, there are poor fluxes on the boundary (boundary locking) and strong oscillations of the reactions forces.

3.2 Methods based on a modification of the weak form

We review and compare some of the most powerful techniques for the imposition of essential boundary conditions with X-FEM. This section presents three methods that overcome this problem: the Lagrange multiplier method, the penalty method and Nitsche's method.

3.2.1 Background on these methods

For the sake of clarity, the following model problem is considered (see in reference [48]). The resolution of the 2D linear elasticity problem represented in Figure 3.1.

3.2.1.1 The elasticity problem

Consider a solid occupying a domain Ω with boundary

$$\Gamma = \Gamma_u \cup \Gamma_t \text{ with } \Gamma_u \cap \Gamma_t = \varnothing \tag{3.1}$$

$$\textit{Equilibrium}: \sigma_{ij,j} + f_i = 0 \ \ in\,\Omega$$
$$\textit{Constitutive law}: \sigma_{ij} = D_{ijkl}\varepsilon_{kl}$$
$$\textit{Strain}-\textit{displacement}: \varepsilon_{ij} = \frac{1}{2}(u_{i,j} + u_{j,i})$$
$$\textit{Dirichlet b.c.}: \ u_i - u_i^{sp} = 0 \quad on\ \Gamma_u$$
$$\textit{Neumann b.c.}: \sigma_{ij} n_j - t_i^{sp} = 0 \quad on\ \Gamma_t$$

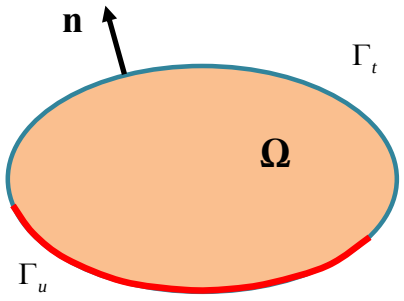


Figure 3.1. *2D linear elasticity problem.*

$u_i = \text{i}^{\text{th}}$ component of the displacement vector **u**
$\sigma_{ij} = ij^{th}$ component of the Cauchy stress
$\varepsilon_{ij} = ij^{th}$ component of strain tensor
$D_{ijkl} =$ elasticity tensor
$t_i^{sp} =$ specified traction vector on Γ_t
$n_i = i^{th}$ component of the unit outward normal

The constrained minimization problem may be stated as

$$\mathbf{u}\left(\in H^1\right) = \arg\min_{\mathbf{w}\in H^1} J(\mathbf{w}) \tag{3.2}$$

subject to the constraint

$$\mathbf{u} - \mathbf{u}^{sp} = 0 \quad on\ \Gamma_u \tag{3.3}$$

where

$$J(\mathbf{w}) = \frac{1}{2}a(\mathbf{w},\mathbf{w}) - l(\mathbf{w})$$

$$where\ a(\mathbf{w},\mathbf{w}) = \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w})^T \mathbf{D}\boldsymbol{\varepsilon}(\mathbf{w}) d\Omega \tag{3.4}$$

$$l(\mathbf{w}) = \int_\Omega \mathbf{w}^T \mathbf{f}\ d\Omega + \int_{\Gamma_t} \mathbf{w}^T \mathbf{t}^{sp} dS$$

The vectors and matrices in 2D are

$$\mathbf{w} = [w_1, w_2]^T$$

$$\boldsymbol{\varepsilon}(\mathbf{w}) = \left[\frac{\partial w_1}{\partial x_1}, \frac{\partial w_2}{\partial x_2}, \left(\frac{\partial w_1}{\partial x_2} + \frac{\partial w_2}{\partial x_1}\right)\right]^T = \boldsymbol{\partial}\mathbf{w} \qquad \boldsymbol{\partial} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix} \tag{3.5}$$

$$\mathbf{f} = [f_1, f_2]$$

$$\mathbf{t}^{sp} = [t_1^{sp}, t_2^{sp}]$$

$$\mathbf{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix} \text{ plane stress;} \quad \mathbf{D} = \frac{E}{(1+\nu)(1-2\nu)}\begin{bmatrix} (1-\nu) & \nu & 0 \\ \nu & (1-\nu) & 0 \\ 0 & 0 & \frac{1-2\nu}{2} \end{bmatrix} \text{ plane strain} \tag{3.6}$$

3.2.1.2 **Penalty method:** implementation

Construct a modified functional

$$J_\alpha(\mathbf{w}) = J(\mathbf{w}) + \frac{\alpha}{2}\int_{\Gamma_u} (\mathbf{w} - \mathbf{u}^{sp})^T (\mathbf{w} - \mathbf{u}^{sp})\, dS \tag{3.7}$$

Solve the unconstrained minimization problem

$$\mathbf{u}\left(\in H^1\right) = \arg\min_{\mathbf{w}\in H^1} J_\alpha(\mathbf{w})\ as\ \alpha \to \infty \tag{3.8}$$

Approximate solution

$$\underbrace{\mathbf{u}_h}_{\in X_h} = \arg\min_{\mathbf{w}_h \in X_h} J_\alpha(\mathbf{w}_h)\ as\ \alpha \to \infty \tag{3.9}$$

X_h is a finite dimensional subspace of H^1

where the discretized modified functional

$$J_\alpha(\mathbf{w}_h) = \underbrace{J(\mathbf{w}_h)}_{Part\ A} + \underbrace{\frac{\alpha}{2}\int_{\Gamma_u} (\mathbf{w}_h - \mathbf{u}^{sp})^T (\mathbf{w}_h - \mathbf{u}^{sp})\, dS}_{Part\ B} \tag{3.10}$$

Now, discretize

$$\mathbf{w}_h = \sum_k \varphi_k \left(\mathbf{w}_h\right)_k = \boldsymbol{\Phi}\mathbf{W}_h \tag{3.11}$$

where

$$\mathbf{W}_h \in \mathbb{R}^N = \text{ vector of nodal unknowns}$$

$$\boldsymbol{\Phi} = \text{Shape function matrix} = \begin{bmatrix} \cdots & \varphi_i & 0 & \cdots \\ & 0 & \varphi_i & \end{bmatrix}$$

$$J(\mathbf{W}_h \in \mathbb{R}^N) = \underbrace{\frac{1}{2}\mathbf{W}_h{}^T \mathbf{A}_h \mathbf{W}_h}_{\textit{Part A}} - \underbrace{\mathbf{W}_h{}^T \mathbf{F}_h}_{\textit{Part B}} \tag{3.12}$$

Part A, proof

$$\text{since } \boldsymbol{\varepsilon}(\mathbf{w}_h) = \partial \mathbf{w}_h = \sum_k (\partial \varphi_k)(\mathbf{w}_h)_k = \sum_k \boldsymbol{\Delta}_k (\mathbf{w}_h)_k = \boldsymbol{\Delta}\mathbf{W}_h$$

$$\textit{where } \mathbf{B}_k = \partial \varphi_k = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 \\ 0 & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} \end{bmatrix} \varphi_k = \begin{bmatrix} \frac{\partial \varphi_k}{\partial x_1} & 0 \\ 0 & \frac{\partial \varphi_k}{\partial x_2} \\ \frac{\partial \varphi_k}{\partial x_2} & \frac{\partial \varphi_k}{\partial x_1} \end{bmatrix} \tag{3.13}$$

$$\text{Hence } a(\mathbf{w}_h, \mathbf{w}_h) = \int_\Omega \boldsymbol{\varepsilon}(\mathbf{w}_h)^T \mathbf{D} \boldsymbol{\varepsilon}(\mathbf{w}_h) d\Omega$$

$$= \mathbf{W}_h{}^T \underbrace{\left(\int_\Omega \boldsymbol{\Delta}^T \mathbf{D} \boldsymbol{\Delta} d\Omega \right)}_{\mathbf{A}_h} \mathbf{W}_h$$

Part B, proof

$$l(\mathbf{w}_h) = \int_\Omega \mathbf{w}_h{}^T \mathbf{f}\, d\Omega + \int_{\Gamma_t} \mathbf{w}_h{}^T \mathbf{t}^{sp} dS$$

$$= \mathbf{W}_h{}^T \left(\int_\Omega \boldsymbol{\Phi}^T \mathbf{f}\, d\Omega \right) + \mathbf{W}_h{}^T \left(\int_{\Gamma_t} \mathbf{w}_h{}^T \mathbf{t}^{sp} dS \right) \tag{3.14}$$

$$= \mathbf{W}_h{}^T \underbrace{\left(\int_\Omega \boldsymbol{\Phi}^T \mathbf{f}\, d\Omega + \int_{\Gamma_t} \mathbf{w}_h{}^T \mathbf{t}^{sp} dS \right)}_{\mathbf{F}_h}$$

We have,

$$J_\alpha(\mathbf{w}_h) = \underbrace{J(\mathbf{w}_h)}_{\textit{Part A}} + \frac{\alpha}{2} \underbrace{\int_{\Gamma_u} (\mathbf{w}_h - \mathbf{u}^{sp})^T (\mathbf{w}_h - \mathbf{u}^{sp})\, dS}_{\textit{Part B}} \tag{3.15}$$

Part B,

using $\mathbf{w}_h = \sum_k \varphi_k (\mathbf{w}_h)_k = \boldsymbol{\Phi}\mathbf{W}_h$ (3.16)

$$\textit{Part B} = \int_{\Gamma_u} (\boldsymbol{\Phi}\mathbf{W}_h - \mathbf{u}^{sp})^T (\boldsymbol{\Phi}\mathbf{W}_h - \mathbf{u}^{sp})\, dS$$

$$= \mathbf{W}_h{}^T \left(\int_{\Gamma_u} \boldsymbol{\Phi}^T \boldsymbol{\Phi} dS \right) \mathbf{W}_h - 2\mathbf{W}_h{}^T \int_{\Gamma_u} \boldsymbol{\Phi}^T \mathbf{u}^{sp} dS + \int_{\Gamma_u} \mathbf{u}^{spT} \mathbf{u}^{sp} dS \tag{3.17}$$

Hence,

$$
\begin{aligned}
J_\alpha(\mathbf{W}_h \in \mathbb{R}^N) &= \underbrace{J(\mathbf{W}_h)}_{\textit{Part A}} \\
&+ \frac{\alpha}{2}\underbrace{\left[\mathbf{W}_h{}^T \left(\int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{\Phi} dS \right) \mathbf{W}_h - 2\mathbf{W}_h{}^T \int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{u}^{sp} dS + \int_{\Gamma_u} \mathbf{u}^{spT} \mathbf{u}^{sp} dS \right]}_{\textit{Part B}} \\
\Rightarrow \frac{\partial J_\alpha}{\partial \mathbf{W}_h} &= \underbrace{\frac{\partial J}{\partial \mathbf{W}_h}}_{\textit{Part A}} + \alpha \underbrace{\left[\left(\int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{\Phi} dS \right) \mathbf{W}_h - \int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{u}^{sp} dS \right]}_{\textit{Part B}} \\
&= \underbrace{\left(\mathbf{A}_h \mathbf{W}_h - \mathbf{F}_h \right)}_{\textit{Part A}} + \alpha \underbrace{\left(\int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{\Phi} dS \right)}_{\hat{\mathbf{A}}_h} \mathbf{W}_h - \alpha \underbrace{\int_{\Gamma_u} \mathbf{\Phi}^T \mathbf{u}^{sp} dS}_{\hat{\mathbf{F}}_h} \\
&= \left(\mathbf{A}_h + \alpha \hat{\mathbf{A}}_h \right) \mathbf{W}_h - \left(\mathbf{F}_h + \alpha \hat{\mathbf{F}}_h \right) = \mathbf{0} \\
&\frac{\partial J_\alpha(\mathbf{W_h})}{\partial \mathbf{W_h}} = 0 \\
&\Rightarrow \underbrace{\left(\mathbf{A}_h + \alpha \hat{\mathbf{A}}_h \right)}_{\tilde{\mathbf{A}}_h} \mathbf{W}_h = \underbrace{\mathbf{F}_h + \alpha \hat{\mathbf{F}}_h}_{\tilde{\mathbf{F}}}
\end{aligned}
\tag{3.18}
$$

Note:

- $\tilde{A}_h$ is symmetric.
- $\tilde{A}_h$ is ill conditioned for large α.
- The formulation is not consistent for trial functions that do not vanish on the Dirichlet boundary.

3.2.1.3 **Penalty method:** problem

The formulation is *not consistent* for trial functions that do not vanish on the Dirichlet boundary. To see this, start with the modified functional

$$
J_\alpha(u_i) = J(u_i) + \frac{\alpha}{2} \int_{\Gamma_u} (u_i - u_i^{sp})^2 dS \tag{3.19}
$$

$$
\begin{aligned}
&\Rightarrow \delta J_\alpha(u_i) = \delta J(u_i) + \alpha \int_{\Gamma_u} \delta u_i (u_i - u_i^{sp}) dS \\
&\textit{with } \delta J(u_i) = \int_\Omega \delta \varepsilon_{ij} D_{ijkl} \varepsilon_{kl} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&\qquad = \int_\Omega \underbrace{\delta u_{i,j} D_{ijkl} u_{k,l}}_{\left(\delta u_i D_{ijkl} u_{k,l}\right)_{,j} - \delta u_i D_{ijkl} u_{k,lj} \text{ (chain rule)}} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&\qquad = \int_\Omega \left(\delta u_i D_{ijkl} u_{k,l} \right)_{,j} d\Omega - \int_\Omega \delta u_i D_{ijkl} u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS
\end{aligned}
\tag{3.20}
$$

Using Green's theorem

$$
\begin{aligned}
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i D_{ijkl} u_{k,l} n_j dS - \int_\Omega \delta u_i D_{ijkl} u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i \underbrace{\sigma_{ij} n_j}_{t_i} dS - \int_\Omega \delta u_i D_{ijkl} u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i t_i dS - \int_{\Gamma_t} \delta u_i t_i^{sp} dS - \int_\Omega \delta u_i \left(D_{ijkl} u_{k,lj} + f_i \right) d\Omega \\
&= \int_{\Gamma_u} \delta u_i t_i dS + \int_{\Gamma_t} \delta u_i \left(t_i - t_i^{sp} \right) dS - \int_\Omega \delta u_i \left(\sigma_{ij,j} + f_i \right) d\Omega \\
\Rightarrow \delta J(u_i) &= \int_{\Gamma_u} \delta u_i t_i dS + \int_{\Gamma_t} \delta u_i \left(t_i - t_i^{sp} \right) dS - \int_\Omega \delta u_i \left(\sigma_{ij,j} + f_i \right) d\Omega
\end{aligned} \tag{3.21}
$$

Thus,

$$
\delta J_\alpha(u_i) = \underbrace{\int_{\Gamma_u} \delta u_i t_i dS}_{\substack{\text{Nothing to balance this} \\ \text{unless } \delta u_i = 0 \text{ on } \Gamma_u}} + \int_{\Gamma_t} \delta u_i \underbrace{\left(t_i - t_i^{sp} \right)}_{=0} dS - \int_\Omega \delta u_i \underbrace{\left(\sigma_{ij,j} + f_i \right)}_{=0} d\Omega + \alpha \int_{\Gamma_u} \delta u_i \underbrace{\left(u_i - u_i^{sp} \right)}_{=0} dS \tag{3.22}
$$

Neumann b.c — Equilibrium eq — Dirichlet b.c

This lack of consistency is a major problem for methods where the trial function space does not vanish on the Dirichlet boundary. We will see that Nitsche's method overcomes this problem. But before that we need to understand the physical interpretation of Lagrange multipliers.

3.2.1.4 **Lagrange multipliers:** basic

Constructed a modified functional

$$
J_{\boldsymbol{\lambda}}(\mathbf{w}) = J(\mathbf{w}) + \int_{\Gamma_u} \boldsymbol{\lambda}^T (\mathbf{w} - \mathbf{u}^{sp}) \, dS \tag{3.23}
$$

Solved the unconstrained minimization problem

$$
\left(\mathbf{u} \in H^1, \boldsymbol{\lambda} \in H^0 \right) = \arg \min_{\mathbf{w}\in H^1, \boldsymbol{\lambda}\in H^0} J_{\boldsymbol{\lambda}}(\mathbf{w}) \tag{3.24}
$$

The problem posed on finite dimensional subspaces

$$
\left(\mathbf{u}_h, \boldsymbol{\lambda}_h \in H^0 \right) = \arg \min_{\mathbf{w}_h, \boldsymbol{\lambda}_h} J_{\boldsymbol{\lambda}}(\mathbf{w}_h) \tag{3.25}
$$

3.2.1.5 **Lagrange multipliers**: implementation

Discretization

$$
\begin{aligned}
\mathbf{w}_h &= \sum_{k=1}^{N} \varphi_k \left(\mathbf{w}_h \right)_k = \boldsymbol{\Phi} \mathbf{W}_h \quad (\mathbf{W}_h \in \mathbb{R}^N) \\
\boldsymbol{\lambda}_h &= \sum_{k=1}^{M} \xi_k \boldsymbol{\lambda}_{h\,k} = \boldsymbol{\Sigma} \boldsymbol{\Lambda}_h \quad (\boldsymbol{\Lambda}_h \in \mathbb{R}^M)
\end{aligned} \tag{3.26}
$$

The modified functional

$$
J_\lambda(\mathbf{W_h}) = J(\mathbf{W_h}) + \boldsymbol{\Lambda}_h^T \left(\mathbf{B}_h \, \mathbf{W}_h - \mathbf{G}_h \right) \tag{3.27}
$$

where

$$J(\mathbf{W}_h) = \frac{1}{2}\mathbf{W}_h{}^T\mathbf{A}_h\mathbf{W}_h - \mathbf{W}_h{}^T\mathbf{F}_h$$
$$\mathbf{B}_h = \int_{\Gamma_u} \mathbf{\Sigma}^T\mathbf{\Phi}\, dS \tag{3.28}$$
$$\mathbf{G}_h = \int_{\Gamma_u} \mathbf{\Sigma}^T\mathbf{u}^{sp}\, dS$$

Minimization

$$\frac{\partial J_\lambda}{\partial \mathbf{W}_h} = \mathbf{A}_h\mathbf{W}_h - \mathbf{F}_h + \mathbf{B}_h{}^T\mathbf{\Lambda}_h = \mathbf{0}$$
$$\frac{\partial J_\lambda}{\partial \mathbf{\Lambda}_h} = \mathbf{B}_h\mathbf{W}_h - \mathbf{G}_h = \mathbf{0} \tag{3.29}$$

In matrix form : Saddle point problem

$$\begin{pmatrix} \mathbf{A_h} & \mathbf{B}_h{}^T \\ \mathbf{B}_h & 0 \end{pmatrix}\begin{pmatrix} \mathbf{w}_h \\ \mathbf{\Lambda}_h \end{pmatrix} = \begin{pmatrix} \mathbf{F}_h \\ \mathbf{G}_h \end{pmatrix} \qquad \begin{matrix} \mathbf{A_h} \in R^{N\times N} \\ \mathbf{B}_h \in R^{M\times N} \end{matrix} \tag{3.30}$$

- Symmetric,
- Well conditioned,
- Higher computational cost as number of unknowns increase,
- May not be positive definite.

3.2.1.6 **Lagrange multipliers**: physical interpretation

Claim: The physical interpretation of the Lagrange multiplier is that it represents the traction at the Dirichlet boundary

$$\lambda_i = -\sigma_{ij}n_j \; on\, \Gamma_u \tag{3.31}$$

Proof:
Start with the modified functional

$$J_{\boldsymbol{\lambda}}(u_i) = J(u_i) + \int_{\Gamma_u} \lambda_i(u_i - u_i^{sp})\, dS$$
$$\Rightarrow \delta J_{\boldsymbol{\lambda}}(u_i) = \underbrace{\delta J(u_i)}_{Part\ A} + \underbrace{\int_{\Gamma_u} \delta\lambda_i(u_i - u_i^{sp})\, dS \;\; + \int_{\Gamma_u} \lambda_i\delta u_i dS}_{Part\ B} \tag{3.32}$$

Lets first look at part A

$$J(u_i) = \frac{1}{2}\int_\Omega \varepsilon_{ij}D_{ijkl}\varepsilon_{kl}d\Omega - \int_\Omega u_i f_i\, d\Omega - \int_{\Gamma_t} u_i t_i^{sp} dS$$
$$\Rightarrow \delta J(u_i) = \int_\Omega \delta\varepsilon_{ij}D_{ijkl}\varepsilon_{kl}d\Omega - \int_\Omega \delta u_i f_i\, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS$$
$$= \int_\Omega \underbrace{\delta u_{i,j}D_{ijkl}u_{k,l}}_{\left(\delta u_i D_{ijkl}u_{k,l}\right)_{,j} - \delta u_i D_{ijkl}u_{k,lj}\ \text{(chain rule)}} d\Omega - \int_\Omega \delta u_i f_i\, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \tag{3.33}$$
$$= \int_\Omega \left(\delta u_i D_{ijkl}u_{k,l}\right)_{,j} d\Omega - \int_\Omega \delta u_i D_{ijkl}u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i\, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS$$

Using Green's Theorem

$$
\begin{aligned}
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i D_{ijkl} u_{k,l} n_j dS - \int_\Omega \delta u_i D_{ijkl} u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i \underbrace{\sigma_{ij} n_j}_{t_i} dS - \int_\Omega \delta u_i D_{ijkl} u_{k,lj} d\Omega - \int_\Omega \delta u_i f_i \, d\Omega - \int_{\Gamma_t} \delta u_i t_i^{sp} dS \\
&= \int_{\Gamma=\Gamma_u\cup\Gamma_t} \delta u_i t_i dS - \int_{\Gamma_t} \delta u_i t_i^{sp} dS - \int_\Omega \delta u_i \left(D_{ijkl} u_{k,lj} + f_i \right) d\Omega \\
&= \int_{\Gamma_u} \delta u_i t_i dS + \int_{\Gamma_t} \delta u_i \left(t_i - t_i^{sp} \right) dS - \int_\Omega \delta u_i \left(\sigma_{ij,j} + f_i \right) d\Omega \\
\Rightarrow \delta J(u_i) &= \int_{\Gamma_u} \delta u_i t_i dS + \int_{\Gamma_t} \delta u_i \left(t_i - t_i^{sp} \right) dS - \int_\Omega \delta u_i \left(\sigma_{ij,j} + f_i \right) d\Omega
\end{aligned}
\tag{3.34}
$$

Consequently,

$$
\begin{aligned}
\delta J_{\boldsymbol{\lambda}}(u_i) &= \underbrace{\delta J(u_i)}_{Part\ A} + \underbrace{\int_{\Gamma_u} \delta\lambda_i (u_i - u_i^{sp}) \, dS \;\; + \int_{\Gamma_u} \lambda_i \delta u_i dS}_{Part\ B} \\
&= \underbrace{\int_{\Gamma_u} \delta u_i t_i dS + \int_{\Gamma_t} \delta u_i \overbrace{\left(t_i - t_i^{sp} \right)}^{Neumann\ b.c.} dS - \int_\Omega \delta u_i \overbrace{\left(\sigma_{ij,j} + f_i \right)}^{Equilibrium\ eq} d\Omega}_{Part\ A} \\
&\quad + \underbrace{\int_{\Gamma_u} \delta\lambda_i \overbrace{(u_i - u_i^{sp})}^{Dirichlet\ b.c.} dS \;\; + \int_{\Gamma_u} \lambda_i \delta u_i dS}_{Part\ B}
\end{aligned}
\tag{3.35}
$$

Hence, we identify $\boxed{\lambda_i = -\sigma_{ij} n_j \; on \; \Gamma_u}$

We may now replace the Lagrange multiplier with its physical interpretation to define the modified functional.

$$
\begin{aligned}
J_{\boldsymbol{\lambda}}(u_i) &= J(u_i) + \int_{\Gamma_u} \lambda_i (u_i - u_i^{sp}) \, dS \\
&= J(u_i) - \int_{\Gamma_u} \sigma_{ij} n_j (u_i - u_i^{sp}) \, dS \\
&= J(u_i) - \int_{\Gamma_u} D_{ijkl} \varepsilon_{kl} n_j (u_i - u_i^{sp}) \, dS \\
&= J(u_i) - \int_{\Gamma_u} D_{ijkl} u_{k,l} n_j (u_i - u_i^{sp}) \, dS
\end{aligned}
\tag{3.36}
$$

Notice that, due to minor symmetry

$$
D_{ijkl} \varepsilon_{kl} = D_{ijkl} \frac{1}{2} (u_{k,l} + u_{l,k}) = D_{ijkl} u_{k,l} \tag{3.37}
$$

The advantage of using the modified functional

$$
J_{\boldsymbol{\lambda}}(u_i) = J(u_i) - \int_{\Gamma_u} D_{ijkl} u_{k,l} n_j (u_i - u_i^{sp}) \, dS \tag{3.38}
$$

Is that the number of equations do not increase? In vector form

$$
J_{\boldsymbol{\lambda}}(\mathbf{u}) = J(\mathbf{u}) - \int_{\Gamma_u} \left(\mathbf{N}\boldsymbol{\sigma} \right)^T (\mathbf{u} - \mathbf{u}^{sp}) \, dS \tag{3.39}
$$

with

$$\mathbf{N}=\begin{bmatrix} n_1 & 0 & n_2 \\ 0 & n_2 & n_1 \end{bmatrix};\qquad \boldsymbol{\sigma}=\begin{bmatrix}\sigma_{11} & \sigma_{22} & \sigma_{12}\end{bmatrix}^T=\mathbf{D}\boldsymbol{\Delta}\mathbf{U}$$
$$J_{\boldsymbol{\lambda}}(\mathbf{u})=J(\mathbf{u})-\int_{\Gamma_u}\left(\mathbf{N}\boldsymbol{\sigma}\right)^T(\mathbf{u}-\mathbf{u}^{sp})\,dS \tag{3.40}$$
$$\Rightarrow J_{\boldsymbol{\lambda}}(\mathbf{U})=J(\mathbf{U})-\int_{\Gamma_u}\mathbf{U}^T\boldsymbol{\Delta}^T\mathbf{D}\mathbf{N}^T(\boldsymbol{\Phi}\mathbf{U}-\mathbf{u}^{sp})\,dS$$

Minimizing

$$\frac{\partial J_{\boldsymbol{\lambda}}(\mathbf{U})}{\partial\mathbf{U}}=\frac{\partial J(\mathbf{U})}{\partial\mathbf{U}}-\underbrace{\left(\int_{\Gamma_u}\boldsymbol{\Delta}^T\mathbf{D}\mathbf{N}^T\boldsymbol{\Phi}dS+\int_{\Gamma_u}\boldsymbol{\Phi}^T\mathbf{N}\mathbf{D}\boldsymbol{\Delta}dS\right)}_{\widehat{\mathbf{A}}}\mathbf{U}$$
$$+\underbrace{\left(\int_{\Gamma_u}\boldsymbol{\Delta}^T\mathbf{D}\mathbf{N}^T\mathbf{u}^{sp}dS\right)}_{\widehat{\mathbf{F}}}=\mathbf{0}$$
$$\Rightarrow\left(\mathbf{A}-\widehat{\mathbf{A}}\right)\mathbf{U}=\left(\mathbf{F}-\widehat{\mathbf{F}}\right) \tag{3.41}$$

- Number of unknowns does not increase,
- System matrix remains symmetric,
- Less accurate.

3.2.1.7 **Nitsche's Method:** modified functional

In order to overcome the inaccuracy of the previous method, we combine with the one of the penalty method. Then, the modified functional is

$$J_{\boldsymbol{N}}(u_i)=J(u_i)-\underbrace{\int_{\Gamma_u}\sigma_{ij}(u)n_j(u_i-u_i^{sp})\,dS}_{\substack{\textit{Lagrange multiplier term}\\ \textit{with the Lagrange multiplier}\\ \textit{replaced by its physical interpretation}}}+\underbrace{\frac{\alpha}{2}\int_{\Gamma_u}(u_i-u_i^{sp})^2dS}_{\substack{\textit{Penalty term enforcing}\\ \textit{the same Dirichlet condition}}} \tag{3.42}$$

where α is a positive constant scalar parameter.

3.2.1.8 **Nitsche's Method:** consistency

Then,

$$\delta J_{\boldsymbol{N}}(u_i)=\delta J(u_i)-\int_{\Gamma_u}\delta\sigma_{ij}n_j(u_i-u_i^{sp})\,dS$$
$$-\int_{\Gamma_u}\sigma_{ij}n_j\delta u_i\,dS+\alpha\int_{\Gamma_u}\delta u_i(u_i-u_i^{sp})dS \tag{3.43}$$

with

$$\delta J(u_i)=\int_{\Gamma_u}\delta u_i t_i dS+\int_{\Gamma_t}\delta u_i\underbrace{\left(t_i-t_i^{sp}\right)}_{=0}dS-\int_{\Omega}\delta u_i\underbrace{\left(\sigma_{ij,j}+f_i\right)}_{=0}d\Omega \tag{3.44}$$

Thus,

$$\begin{aligned}\delta J_{\boldsymbol{N}}(u_i) &= \underbrace{\int_{\Gamma_u} \delta u_i t_i dS}_{\delta J(u_i)} - \int_{\Gamma_u} \delta\sigma_{ij} n_j (u_i - u_i^{sp})\, dS \\ &\quad - \int_{\Gamma_u} \sigma_{ij} n_j \delta u_i \, dS + \alpha \int_{\Gamma_u} \delta u_i (u_i - u_i^{sp}) dS \\ &= \int_{\Gamma_u} \delta u_i \underbrace{\left(t_i - \sigma_{ij} n_j\right)}_{=0} dS - \int_{\Gamma_u} \delta\sigma_{ij} n_j \underbrace{(u_i - u_i^{sp})}_{=0}\, dS \\ &\quad + \alpha \int_{\Gamma_u} \delta u_i \underbrace{(u_i - u_i^{sp})}_{=0} dS\end{aligned} \tag{3.45}$$

The new terms in the r.h.s. are added to ensure consistency of the weak form. So, Nitsche's method restores consistency in the formulation unlike the penalty method.

3.2.2 Penalty method

Assume boundary conditions in "standard form"

$$\left[\tilde{C}\right](D) = (D_o) \tag{3.46}$$

Define an *m* by *m* "penalty matrix"

$$[\kappa] = diag\left(\kappa_1, \kappa_2, ..., \kappa_m\right); \kappa_i \gg \max\left|K_{ij}\right| \tag{3.47}$$

Modify the global problem as follows

$$\left\{[\mathbf{K}] + \left[\tilde{\mathbf{C}}\right]^T [\boldsymbol{\kappa}] \left[\tilde{\mathbf{C}}\right]\right\}(\mathbf{D}) = (\mathbf{F}) + \left[\tilde{\mathbf{C}}\right]^T [\boldsymbol{\kappa}](\mathbf{D}_o) \tag{3.48}$$

The idea is to enforce a displacement boundary condition *approximately* by changing [**K**] and (**F**).

For an interpolation with consistency of order *p* and discretization measure *h* the best error estimate gives a rate of convergence of order $h^{\frac{2p+1}{3}}$ in the energy norm, provided that the penalty α is taken to be of order $h^{\frac{2p+1}{3}}$. In the linear case, it corresponds to the optimal rate of convergence in energy norm. For order $p \geq 2$, the lack of optimality in the rate of convergence is a direct consequence of the lack of consistency of the weak formulation. The choice of the penalty α to maintain the optimal rate of convergence in L^2 norm and the ill-conditioning of the system are commented for a particular problem.

The penalty method can also be obtained from the minimization of the discrete version of the energy function.

Obviously, the situation gets worse for denser discretizations, which need larger penalty parameters. The ill-conditioning of the matrix reduces the applicability of the penalty method. The penalty parameter α

is a positive scalar constant that must be large enough in order to impose the essential boundary condition with the desired accuracy. As previously observed with the Lagrange multiplier method, the penalty method is easily applicable to all kind of problems.

Notes on the penalty method:

- Very easy to implement: no re-numbering, no transformations to apply,
- Does not eliminate dof., so no reduction in bandwidth.

Assigning the penalty numbers κ_i can be tricky:

- Too low => poor approximation to boundary condition.
- Too high => can create numerical problems *(e.g.*, locking, ill-conditioning, …).

The penalty method presents two clear advantages: (i) the dimension of the system is not increased and (ii) the matrix in the resulting system, is symmetric and positive definite, provided that $\boldsymbol{K}$ is symmetric and α is large enough.

However, the penalty method has also two important drawbacks: the Dirichlet boundary condition is weakly imposed (the parameter α controls how well the essential boundary condition is ensured) and the matrix is usually ill-conditioned (the condition number increases with α).

3.2.3 Lagrange multiplier method

Noted that, Lagrange multipliers can be interpreted physically as constraint forces. The main idea is to add extra dofs into the problem, and use these dofs to enforce the boundary conditions.

There are several possibilities for the choice of the interpolation space for the Lagrange multiplier λ. Therefore, the Lagrange multiplier method is, in principle, general and easily applicable to all kind of problems. In fact, there is no need to know the weak form with Lagrange multiplier, it is sufficient to define the discrete energy functional, i.e. compute $\boldsymbol{K}$ and $\boldsymbol{f}$, and the restrictions due to the boundary conditions, $Au = b$, in order to determine the system of eqs. Advantage is very effective at handling multi-point constraints. However, the main disadvantages of the Lagrange multiplier method are:

- The dimension of the resulting system of equations is increased. More dof., it will take longer solution time.
- Even for $\boldsymbol{K}$ symmetric and semi-positive definite, the global matrix is symmetric but it is no longer positive definite. Therefore, standard linear solvers for symmetric and positive definite matrices cannot be used.
- More crucial is the fact that the system of eqs and the weak problem induce a saddle point problem which precludes an arbitrary choice of the interpolation space for u and λ. The discretization of the multiplier λ must be accurate enough in order to obtain an acceptable solution, but the resulting system of equations turns out to be singular if the number of Lagrange multipliers λ_i is too large. In fact, the interpolation spaces for the Lagrange multiplier λ and for the principal unknown u must verify an inf-sup condition, known as the Ladygenskaya-Babuška-Brezzi (LBB) stability condition, in order to ensure the convergence of the approximation. In the X-FEM, it is trivial to choose the approximation for the Lagrange multiplier to verify the LBB condition.

The first two disadvantages can be neglected in front of the versatility and straightforward implementation of the method. However, while in the extended finite element method it is trivial to choose the interpolation for the Lagrange multiplier in order to verify the LBB stability condition and to impose accurate essential boundary conditions.

3.2.4 Nitsche's method

In this situation, Nitsche's method represents an interesting alternative for the weak imposition of essential boundary conditions. Nitsche's method is a classical method for imposing essential boundary conditions weakly. Unlike the penalty method, it is consistent with the original differential equation. The strong point of Nitsche's method is that it retains the convergence rate of the underlying finite element method, whereas the standard penalty method either requires a very large penalty parameter, destroying the condition number of the resulting matrix problem, or, in case the condition number is to be retained, is limited to first order energy-norm accuracy. The discretization of the Nitsche's weak form leads to a system of equations with the same size as $\boldsymbol{K}$ and whose matrix is symmetric and positive definite, provided that $\boldsymbol{K}$ is symmetric and α is large enough. Although, as in the penalty method, the condition number of this matrix increases with parameter α, in practice not very large values are needed in order to ensure convergence and a proper implementation of the boundary condition. The matrix condition number is not a real problem for this method.

Remark: Nitsche's method can be interpreted as a consistent improvement of the penalty method. The penalty weak form is not consistent. The only problem of Nitsche's method is the deduction of the weak form. The generalization of the implementation for other problems is not as straightforward as for the method of Lagrange multipliers or for the penalty method. The weak form and the choice of parameter α depends not only on the partial differential equation, but also on the essential boundary condition to be prescribed. For increasing values, α plays the role of a penalty parameter, giving more weight to the verification of the boundary condition and, therefore, affecting to the solution in the rest of the domain. The great advantage of Nitsche's method is that parametric tuning can be done with only one scalar parameter α, in front of the difficult choice of the interpolation space for the Lagrange multiplier.

3.2.5 Conclusions

The applicability of the penalty method is reduced due to the possible ill-conditioning problems, specially when refined discretizations are needed. The Lagrange multiplier method and the penalty method present similar properties. The advantage of Nitsche's method is that it requires only the choice of a scalar parameter, in front of the choice of the interpolation space for the Lagrange multiplier. For instance, the choice of the position of the collocation points in the Lagrange multiplier method can be a difficult task for irregular distributions of particles. However, it is fair to recall that the Lagrange multiplier method is easily applicable for the implementation of all sort of linear boundary constraints in a large variety of problems. The penalty method and Nitsche's method require only the choice of one scalar parameter. The applicability of the penalty method is reduced due to the ill-conditioning of the resulting matrix and the lack of consistency of the weak formulation. As an alternative, Nitsche's method introduces new terms in the weak form in order to maintain consistency and coercivity of the bilinear form. Moreover, moderate values of the scalar parameter α (α is a large enough constant which ensures the coercivity of the bilinear form) provide good results, avoiding the ill-conditioning problem of the penalty method. Therefore, Nitsche's method represents an interesting alternative to the widely used Lagrange multiplier method, mainly in those problems where the selection of an appropriate interpolation for the multiplier turns out to be a serious problem.

Therefore, although imposing boundary constraints is straightforward with the Lagrange multiplier method, the applicability of this method in particular cases can be clearly reduced due to the difficulty in the selection of a proper interpolation space for the Lagrange multiplier.

In summary, imposition of constraints

- Penalty method
  - Assume a large penalty parameter.
  - Matrix problem is ill-conditioned.
  - Where to integrate the penalty term and classical drawbacks (choice of the penalty parameter and degraded conditioning).
- Lagrange multipliers
  - The Lagrange parameter is an unknown.
  - A "saddle point problem" results which is symmetric and well-conditioned. However, the problem is indefinite.
  - Physical interpretation of the Lagrange multipliers.
  - Difficulties to choose the correct space.
  - Advantage as reuse of classical contact algorithms.
- Nitsche's method
  - Restores consistency in the penalty formulation.
  - Difficulties of determination of a parameter to insure the stability of the system (inf-sup related in fact). Hard to extend to non-linear bulk behavior. Not easy to reuse all classical contact algorithms.

3.3 Lagrange multiplier approach (1D Version)

Physical meaning of the Lagrange multiplier: jump of the gradient (heat flux).
Matrix form

$$\begin{bmatrix} K & G \\ G^T & 0 \end{bmatrix} \begin{Bmatrix} u \\ \lambda \end{Bmatrix} = \begin{Bmatrix} f \\ q \end{Bmatrix} \tag{3.49}$$

where $G_{JK} = -\Phi_J(x_K)$ and $q_K = -\bar{u}(x_K)$

Note that: x_k are points on the interface

3.4 The choice of the Lagrange Multiplier space

The results obtained with this Lagrange multiplier space are given in Figure 3.2 for a nodal integration. It is clear from these results that locking occurs. The suboptimal convergence rate of the energy error denotes an over-constrained primal variable space. As shown in reference Ji and H, Dolbow 2004 [15], severe oscillations affect the Lagrange multiplier field (see Figure 3.3).

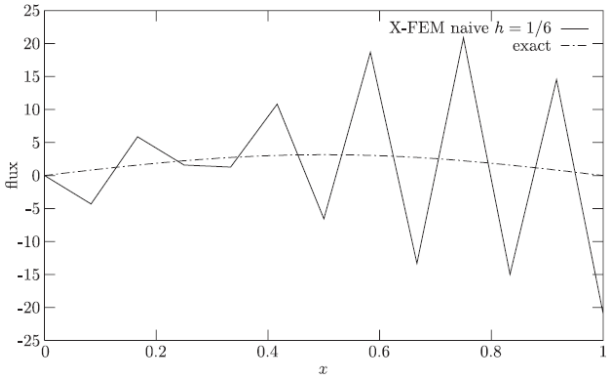


Figure 3.2. *Exact and approximated flux with the naive X-FEM approach taken from [29].*

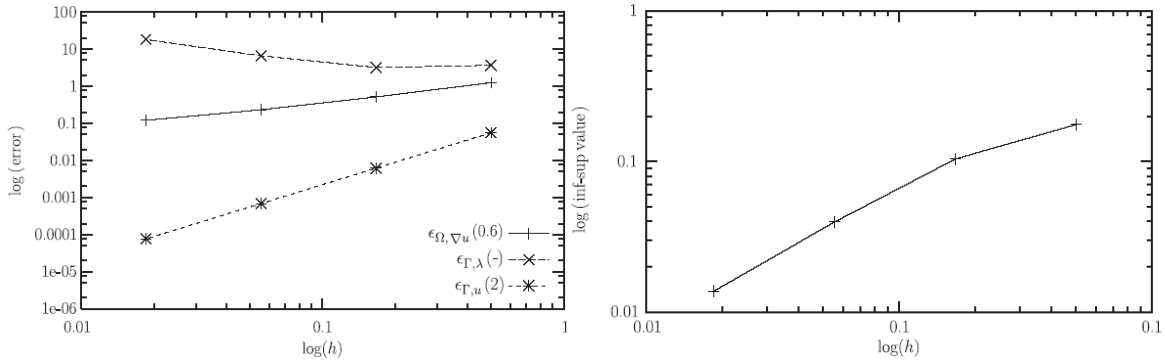


Figure 3.3. *Results for non-matching structured meshes using a full Lagrange multiplier space: convergence of the errors and inf-sup parameter taken from [29].*

To further illustrate why this 'naive' approach fails and the locking issue, we shall see the two-element scalar problem shown in Figure 3.4 [29].

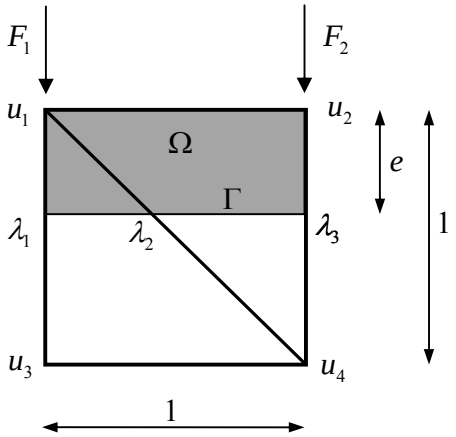


Figure 3.4. *Two-element scalar problem [29].*

The Lagrange multipliers solution is

$$\begin{aligned}\lambda_1 &= F \\ \lambda_2 &= F + \frac{1}{(1-e)}[F] \\ \lambda_3 &= F - \frac{1}{(1-e)^2}[F]\end{aligned} \tag{3.50}$$

where $F = F_1 + F_2$ and $[F] = F_1 - F_2$. We observe the second and third Lagrange multipliers become unbounded when the interface reaches the bottom layer of elements, especially λ_3 whose support drops to zero as e tends to one. Moreover, oscillations occur: the effect of $[F]$ is positive for λ_2 and negative for λ_3.

If we set $\lambda_2 = \lambda_3$, we obtain the Lagrange multiplier below which still exhibit oscillations.

$$\begin{aligned}\lambda_1 &= F - \frac{(2e^3 - 9e^2 + 14e - 8)}{(4e^3 - 12e^2 + 7e + 4)}\frac{1}{e}[F] \\ \lambda_2 &= \lambda_3 \\ \lambda_3 &= F - \frac{(2e^2 - 5e + 4)}{(4e^3 - 12e^2 + 7e + 4)}[F]\end{aligned} \tag{3.51}$$

On the other hand, if we set $\lambda_2 = (1-e)\lambda_1 + e\lambda_3$ (linear variation over Γ) then the oscillations disappear. We obtain

$$\begin{aligned}\lambda_1 &= F + \frac{(2e^3 - 8e^2 + 9e - 4)}{(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4)}[F] \\ \lambda_2 &= (1-e)\lambda_1 + e\lambda_3 \\ \lambda_3 &= F - \frac{(2e^3 - 8e^2 + 9e - 4)}{(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4)}[F]\end{aligned} \tag{3.52}$$

3.5. Algorithm to design the Lagrange multiplier space

Lagrange multipliers were considered for the first time within the concept of X-FEM in (Dolbow, Moës, and Belytschko 2001). In (Ji and Dolbow 2004), the authors illustrate through numerical experiments that a naive choice for the Lagrange multiplier space, i.e., a piecewise linear ansatz space on the interface with a degree of freedom at each node obtained by cutting the interface with the edges of the two dimensional mesh of the primal variable, yields oscillatory multipliers on the boundary. This oscillatory effect is referred as boundary locking. From the mathematical point of view, these oscillations result from a non-uniform but mesh dependent inf-sup condition. Roughly speaking this means that the Lagrange multiplier space is locally too rich, and as a consequence the constant in the inf-sup condition tends to zero if the mesh-size tends to zero.

In order to improve the approach, a first effort was done in (Moës, Béchet, and Tourbier 2006) where a reduced Lagrange multiplier space has been proposed. Although the algorithm to define the Lagrange multiplier space passes a numerical inf-sup test, it is quite complex. Numerically, the Chapelle-Bathe test is used quite often (Chapelle and Bathe 1993; El-Abbasi and Bathe 2001) to verify the inf-sup condition. This test reduces to the computation of eigenvalues for a sequence of meshes with increasing density.

A second algorithm was then proposed in (Geniaut, Massin, and Moës 2007), and it was extended for large sliding (Nistor, Guiton, Massin, Moës, and Geniaut 2008). The strategy used to build the Lagrange multiplier space in the second algorithm is quite easy to grasp. We consider the nodes on each side of the interface. These nodes are tied together across the interface by cut edges. As a subset of these cut edges, we define the set of vital edges as the minimum set of edges able to connect the nodes on each side of the interface. These vital edges and only these will hold a Lagrange multiplier degree of freedom. This second algorithm showed slight improvement in terms of accuracy compared to the first one. The major issue in implementing the second algorithm is that finding the vital edges requires to solve a global problem. Although both approaches perform numerically rather well and no oscillations can be observed, no theoretical analysis of the stability exists.

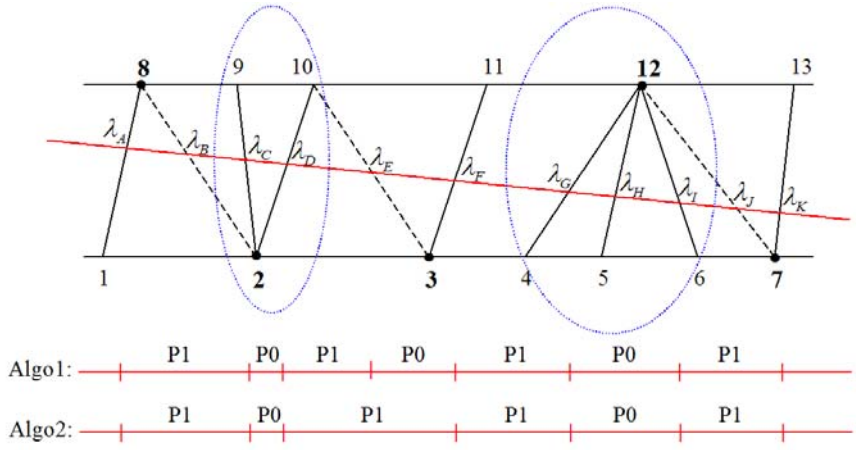


Figure 3.5. *Example of edges cut by an interface. Dotted edges are non-vital, groups of connected vital edges are circled. The final multiplier approximation (PO or P1) is plotted on the interface for first and second algorithms (taken from [49]).*

In this thesis, we use a new algorithm (see Appendix B.3 and B.4), was introduced in E. Béchet, N. Moës, B. Wohlmuth 2008, which allowing a local construction of the Lagrange multiplier space while improving the accuracy of the computed fields. The originality of this approach with regard to (Moës, Béchet, and Tourbier 2006) lies in the use of the trace of primary shape functions defined on the domain, and a simplified procedure to define the mesh on the interface. The proper design of the Lagrange multiplier space is guided by the inf-sup condition (Babuška 1973).

More details about these three algorithms are described in Appendix B: Algorithm to design the Lagrange Multiplier space.

3.6 Definition of norms and error norms

In practice, we shall solve the eigen-problem. If the inf-sup condition is satisfied, optimal rates of convergence are expected. We shall check these rates on three relative errors, namely the error in the 'energy' norm

$$\varepsilon_{\Omega,\nabla u} = \sqrt{\left(\frac{\int_\Omega \nabla\left(u^h - u^{ex}\right) \cdot \nabla\left(u^h - u^{ex}\right) d\Omega}{\int_\Omega \nabla u^{ex} \cdot \nabla u^{ex} d\Omega} \right)} \tag{3.53}$$

The L^2 error norm measuring the accuracy of the Dirichlet boundary condition

$$\varepsilon_{\Gamma,u} = \sqrt{\left(\frac{\int_\Gamma \left(u^h - u^{ex}\right)^2 d\Gamma}{\int_\Gamma \left(u^{ex}\right)^2 d\Gamma} \right)} \tag{3.54}$$

And, finally, the error norm on the Lagrange multiplier

$$\varepsilon_{\Gamma,\lambda} = \sqrt{\left(\frac{\int_\Gamma \left(\lambda^h - \nabla_n u^{ex}\right)^2 d\Gamma}{\int_\Gamma \left(\nabla_n u^{ex}\right)^2 d\Gamma} \right)} \tag{3.55}$$

where ∇ is the gradient operator and ∇_n is its projection on the outward normal on the boundary.

This computation will verify that we have a convergence in energy for each method. In this section the energy is computed with the Eq. (3.56).

$$E = \frac{1}{2} \underline{u}^t K \underline{u} \tag{3.56}$$

This implementation of Eq. (3.56) is simple. This will give equivalent results to the strain energy error computed by Eq. (3.57).

$$Error = \sqrt{\int_{\Omega} \left(\underline{\varepsilon}_{exact}^{T} - \underline{\varepsilon}_{comp}^{T}\right) \underline{\underline{\underline{\underline{C}}}} \left(\underline{\varepsilon}_{exact} - \underline{\varepsilon}_{comp}\right) d\Omega} \tag{3.57}$$

3.7 LBB (inf-sup) condition

These approximations are chosen in one part for the displacement and the other part for the contact pressures are seemly not satisfy the inf-sup condition in all cases. The failure to comply with the Ladygenskaya-Babuška-Brezzi (LBB) condition causes the oscillations of contact pressures, a phenomenon similar to that encountered in incompressibility. Physically, in the case of contact, it is to seek to impose too many contact points of the interface, making the system hyperstatic. To release it, we must restrict the space of Lagrange multipliers, as is done in [29] for Dirichlet conditions with X-FEM. The algorithm proposed by Moës [29] to reduce the oscillations is extended to 3D case. Its purpose is to impose relations of equality or linear relations between Lagrange multipliers. This algorithm tends to impose more equal relationships as linear relations, so the linear approximation of departure is a little worse. This algorithm has been improved to make it more physical and more effective.

$$\inf_{q^h \in Q^h} \sup_{v^h \in V^h} \frac{\int_{\Omega} q^h div v^h d\Omega}{\left\| v^h \right\| \left\| q^h \right\|} \geq \beta > 0 \tag{3.58}$$

with β independent of h

The value of β is simply the min of μ in the following generalized eigen-value problem.

$$K_{up}^{T} M_{uu}^{-1} K_{up} v = \mu^2 M_{pp} v \tag{3.59}$$

The stability of β with respect to h is checked on a sequence of meshes.

3.8. Saddle point problem

3.8.1 Lagrange Multipliers for saddle point problem

$$\begin{pmatrix} \mathbf{A_h} & \mathbf{B^T} \\ \mathbf{B} & 0 \end{pmatrix} \begin{pmatrix} \mathbf{w}_h \\ \boldsymbol{\lambda}_h \end{pmatrix} = \begin{pmatrix} \mathbf{F} \\ \mathbf{G} \end{pmatrix} \quad \begin{matrix} \mathbf{A_h} \in R^{N \times N} & \mathbf{w_h} \in R^N \\ \mathbf{B} \in R^{M \times N} & \boldsymbol{\lambda}_\mathbf{h} \in R^M \end{matrix} \tag{3.60}$$

$$\mathbf{S} = \begin{pmatrix} \mathbf{A_h} & \mathbf{B^T} \\ \mathbf{B} & 0 \end{pmatrix}$$

Observations [48]:

- The matrix S is symmetric,
- The matrix S is usually not ill-conditioned,
- The matrix S is indefinite (i.e., can have eigenvalues that are zero),
- The system has a unique solution for w_h if A_h is symmetric positive definite.

- The system has a unique solution for l_h if anyone of the following (equivalent) statements holds good
  - Statement 1: B^T has a trivial null space.
  - Statement 2: The inf-sup condition is satisfied

$$\beta = \inf_{\substack{\mathbf{q}\in\mathbf{R}^M \\ \mathbf{q}\neq\mathbf{0}}} \sup_{\substack{\mathbf{v}\in\mathbf{R}^N \\ \mathbf{v}\neq\mathbf{0}}} \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} > 0 \tag{3.61}$$

  - Statement 3: is $BA^{-1}B^T$ symmetric positive definite.
- The system allows very attractive iterative solution procedures.
- When a large number of constraints are imposed, solution cost increases considerably (since we have to solve a (N+M)x(N+M) system).

Let $\begin{pmatrix} w_1 \\ \lambda_1 \end{pmatrix}$ and $\begin{pmatrix} w_2 \\ \lambda_2 \end{pmatrix}$ be two solutions of the above equation (we will find the conditions for which it is not possible to have two such solutions)

The difference solution

$$\begin{pmatrix} \Delta\mathbf{w} \\ \Delta\boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{w}_1 - \mathbf{w}_2 \\ \boldsymbol{\lambda}_1 - \boldsymbol{\lambda}_2 \end{pmatrix} \tag{3.62}$$

Satisfies the homogeneous equation

$$\begin{pmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & 0 \end{pmatrix} \begin{pmatrix} \Delta\mathbf{w} \\ \Delta\boldsymbol{\lambda} \end{pmatrix} = \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix} \tag{3.63}$$

Can be written as two sets of equations :

$$\begin{aligned} \mathbf{A}\Delta\mathbf{w} + \mathbf{B}^T\Delta\boldsymbol{\lambda} &= \mathbf{0} \quad (*) \\ \mathbf{B}\Delta\mathbf{w} &= \mathbf{0} \quad (**) \end{aligned} \tag{3.64}$$

Premultiplying (3.64 *) by $\Delta\mathbf{w}^T$ and (3.64 **) by $\Delta\boldsymbol{\lambda}^T$ and subtracting (3.64 **) from (3.64 *)

$$\Delta\mathbf{w}^T\mathbf{A}\Delta\mathbf{w} = 0 \tag{3.65}$$

If A is positive definite (on the set of vectors which lie in the null space of B), then we can say that the above implies

$$\Delta\mathbf{w} = 0 \quad i.e. \ \mathbf{w}_1 = \mathbf{w}_2 \tag{3.66}$$

The next question is, what is the condition for λ to be unique?

Go back to equation (3.64*) with $\Delta\mathbf{w} = 0$

$$\begin{aligned} &\mathbf{A}\Delta\mathbf{w} + \mathbf{B}^T\Delta\boldsymbol{\lambda} = \mathbf{0} \\ &\Rightarrow \mathbf{B}^T\Delta\boldsymbol{\lambda} = \mathbf{0} \end{aligned} \tag{3.67}$$

If B^T has a trivial null space, then

$$\Delta\boldsymbol{\lambda} = 0 \quad i.e. \ \boldsymbol{\lambda}_1 = \boldsymbol{\lambda}_2 \tag{3.68}$$

The following statements are identical:

- Statement 1: B^T has a trivial null space.
- Statement 2: The inf-sup condition is satisfied

$$\beta = \inf_{\substack{\mathbf{q}\in\mathbf{R}^M \\ \mathbf{q}\neq\mathbf{0}}} \sup_{\substack{\mathbf{v}\in\mathbf{R}^N \\ \mathbf{v}\neq\mathbf{0}}} \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} > 0 \tag{3.69}$$

3.8.2 Inf-sup condition for saddle point problem

The inf-sup parameter:

$$f(\mathbf{v},\mathbf{q}) = \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} \tag{3.70}$$

Using Cauchy-Schwartz inequality

$$f(\mathbf{v},\mathbf{q}) = \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} \geq \frac{\|\mathbf{v}\|\|\mathbf{B}^T\mathbf{q}\|}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} = \frac{\|\mathbf{B}^T\mathbf{q}\|}{\sqrt{\mathbf{q}^T\mathbf{q}}} \quad \because \|\mathbf{v}\| = \sqrt{\mathbf{v}^T\mathbf{v}} \tag{3.71}$$

Hence

$$\sup_{\substack{\mathbf{v}\in R^N \\ \mathbf{v}\neq 0}} f(\mathbf{v},\mathbf{q}) = \frac{\|\mathbf{B}^T\mathbf{q}\|}{\sqrt{\mathbf{q}^T\mathbf{q}}} = \frac{\sqrt{\mathbf{q}^T\mathbf{B}\mathbf{B}^T\mathbf{q}}}{\sqrt{\mathbf{q}^T\mathbf{q}}} \tag{3.72}$$

Now consider the following eigenvalue problem

$$\mathbf{B}\mathbf{B}^T\mathbf{x} = \lambda\mathbf{x} \tag{3.73}$$

The matrix BB^T is symmetric positive definite if B^T is full rank and therefore it will have M positive eigenvalues. If B^T is full rank, then the inf-sup condition is satisfied with

$$\inf_{\substack{\mathbf{q}\in R^M \\ \mathbf{q}\neq 0}} \sup_{\substack{\mathbf{v}\in R^N \\ \mathbf{v}\neq 0}} f(\mathbf{v},\mathbf{q}) = \frac{\sqrt{\mathbf{q}^T\mathbf{B}\mathbf{B}^T\mathbf{q}}}{\sqrt{\mathbf{q}^T\mathbf{q}}} = \sqrt{\lambda_{\min}} = \beta \tag{3.74}$$

where $\lambda_{\min}$ is the minimum eigenvalue of BB^T

3.8.3 Example

Consider the matrix

$$\mathbf{S} = \begin{pmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & 0 \end{pmatrix}$$
$$\mathbf{A} = \begin{pmatrix} 1 & 4 & 3 \\ 2 & 7 & 9 \\ 3 & 4 & 7 \end{pmatrix}; \qquad \mathbf{B}^T = \begin{pmatrix} \sigma_1 & 0 \\ 0 & \sigma_2 \\ 0 & 0 \end{pmatrix} \tag{3.75}$$

with $\sigma_1 > \sigma_2 > 0$

$$\sup_{\substack{\mathbf{v}\in R^3 \\ \mathbf{v}\neq 0}} \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} = \frac{\|\mathbf{B}^T\mathbf{q}\|}{\sqrt{\mathbf{q}^T\mathbf{q}}} = \frac{\sqrt{\mathbf{q}^T\mathbf{B}\mathbf{B}^T\mathbf{q}}}{\sqrt{\mathbf{q}^T\mathbf{q}}} \tag{3.76}$$

Now, for arbitrary $q = [q_1, q_2]^T$

$$\sup_{\substack{\mathbf{v}\in R^3 \\ \mathbf{v}\neq 0}} \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} = \frac{\sqrt{\mathbf{q}^T\mathbf{B}\mathbf{B}^T\mathbf{q}}}{\sqrt{\mathbf{q}^T\mathbf{q}}} = \sqrt{\frac{\sigma_1^2 q_1^2 + \sigma_2^2 q_2^2}{q_1^2 + q_2^2}}$$
$$= \sqrt{\frac{\sigma_1^2 q^2 + \sigma_2^2}{q^2 + 1}} \quad \textit{where} \quad q = q_1/q_2 \tag{3.77}$$

The expression within the radical sign is a symmetric positive function of the variable 'q' whose minimum value is σ_2.

Therefore,

$$\inf_{\substack{\mathbf{q}\in R^2 \\ \mathbf{q}\neq 0}} \sup_{\substack{\mathbf{v}\in R^3 \\ \mathbf{v}\neq 0}} \frac{\mathbf{v}^T\mathbf{B}^T\mathbf{q}}{\sqrt{\mathbf{v}^T\mathbf{v}}\sqrt{\mathbf{q}^T\mathbf{q}}} = \min\left(\sqrt{\frac{\sigma_1^{\,2}q^2 + \sigma_2^{\,2}}{q^2+1}}\right) = \sigma_2 \tag{3.78}$$

Hence, the inf-sup condition provides a very practical means of evaluating whether B^T has a null space in this example, σ_2 needs to be positive.

Chapter 4
Numerical examples

In this thesis, the application of the X-FEM to impose kinematic conditions along interfaces problems in two-dimensional solid mechanics is explored. During this thesis many computations have been made. The main computations made during this report are explained below. In the following section you will see several case description. Mainly it was to study the influence of some parameters.
In this research, various applications of the X-FEM were used and applied to 2D problems. Two 2D numerical examples are used to illustrate the methods described in previous sections for the imposition of essential boundary conditions. Let's see these following cases, numerical examples are given:

- **Example 1**: Numerical simulation of an elastic plate of dimensions (-1,1) x (-1,1) cut by a circle has the radius equal to R= 0.4 and then glue the parts back together.

- **Example 2**: Numerical simulation of an elastic plate of dimensions (-1,1) x (-1,1) with a circular hole (radius equal to 0.4) cut by a circle has the radius equal to R= 0.7 and then glue the parts back together.

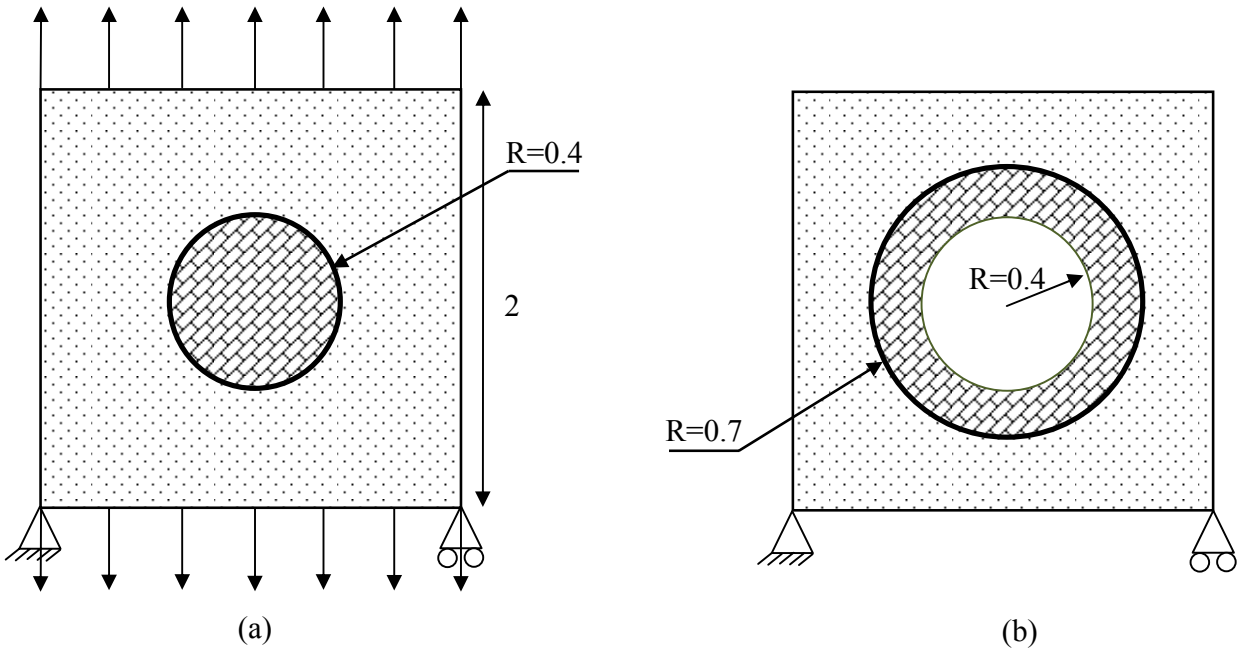


Figure 4.1. *The unit-square model problem with geometry and loading of two examples: (a) example 1; (b) example 2 (plate with a hole).*

The plate lies in a square domain (-1,1)x(-1,1). The center circle is located at (0,0) and has for radius a variable R=0.4. The material is metal for which the Young's modulus is taken equal to 1 and the poisson coefficient equal to 0.3. The bottom left side is clamped. They are subjected to two types of loading. Firstly, we apply the loading which represented in Figure 4.1. In the other case, we consider the problem of a plate with a traction-free circular hole submitted to uniaxial tension.

Plate with hole in tension:

Suppose that the plate is a homogeneous isotropic elastic body.

where a is the radius of the hole and (r,θ) is a polar coordinate system, the origin of which is located at the center of the hole.

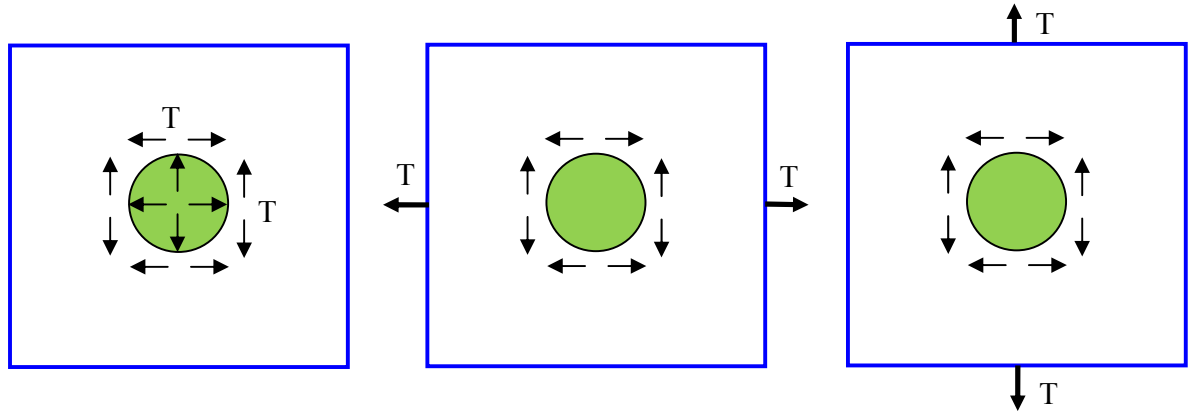


Figure 4.2. *(a) Pressure in a hole; (b) and (c) traction-free circular hole submitted to uniaxial tension.*

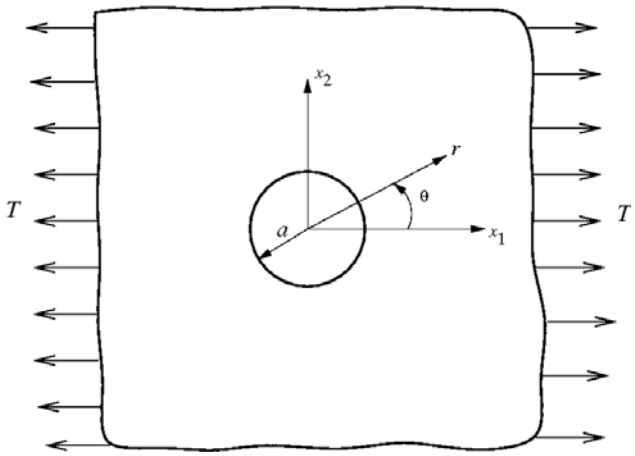


Figure 4.3. *An elastic plate with a circular hole under uniaxial tension.*

This includes the stresses around a circular hole. The expressions for stress components into polar coordinates, using the relations

$$x = r\cos(\theta);\quad y = r\sin(\theta)$$
$$\theta = \text{atan2}(y, x);\quad r = \sqrt{x^2 + y^2}$$

There exists an exact solution for the stresses (Timoshenko and Goodier, 1970)

$$\sigma_r = \frac{T}{2}\left(1 - \frac{a^2}{r^2}\right) + \frac{T\cos(2\theta)}{2}\left(\frac{3a^4}{r^4} - \frac{4a^2}{r^2} + 1\right)$$
$$\sigma_t = \frac{T}{2}\left(1 + \frac{a^2}{r^2}\right) - \frac{T\cos(2\theta)}{2}\left(\frac{3a^4}{r^4} + 1\right)$$
$$\tau_{rt} = \frac{T\sin(2\theta)}{2}\left(\frac{3a^4}{r^4} - \frac{2a^2}{r^2} - 1\right)$$

$$\sigma = \begin{bmatrix} \sigma_{11} & \sigma_{12} & 0 \\ \sigma_{21} & \sigma_{22} & 0 \\ 0 & 0 & \sigma_{33} \end{bmatrix}$$

$$\sigma_{11} = \sigma_r \cos^2(\theta) + \sigma_t \sin^2(\theta) - 2\tau_{rt}\sin(\theta)\cos(\theta)$$
$$\sigma_{12} = (\sigma_r - \sigma_t)\sin(\theta)\cos(\theta) + \tau_{rt}\left(\cos^2(\theta) - \sin^2(\theta)\right)$$
$$\sigma_{21} = \sigma_{12}$$
$$\sigma_{22} = \sigma_r \sin^2(\theta) + \sigma_t \cos^2(\theta) + 2\tau_{rt}\sin(\theta)\cos(\theta)$$
$$\sigma_{33} = \nu \cdot (\sigma_{11} + \sigma_{22})$$

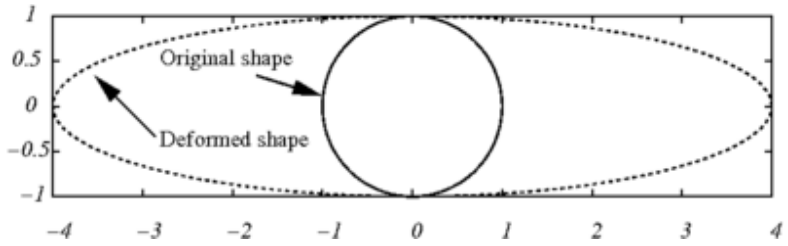


Figure 4.4. *Deformation of the hole under tension*.

In order to investigate the behaviour of the new approach, we computed the L^2 norm for each simulation.

The exact energy norm and the error in energy norm are defined in chapter 2.

For instance, calculate analytical the exact energy norm on Lagrange multiplier is shown the value of 1.482941286

For a radius = 0.7

$$\int \sin^2 axdx = \frac{x}{2} - \frac{\sin 2ax}{4a}$$

$$err_{ext} = \left(r\int_0^{2\pi} \sin^2\theta d\theta = \frac{\theta}{2} - \frac{\sin 2\theta}{4} \right)^{1/2}$$

$$err_{ext} = \left(0.7\int_0^{2\pi} \sin^2\theta d\theta \right)^{1/2} = \left(0.7\left[\frac{\theta}{2} - \frac{\sin 2\theta}{4} \right]_0^{2\pi} \right)^{1/2}$$

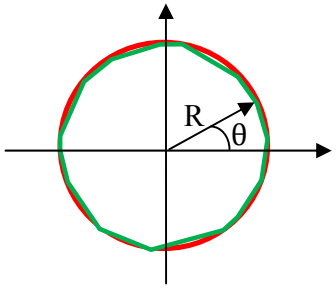
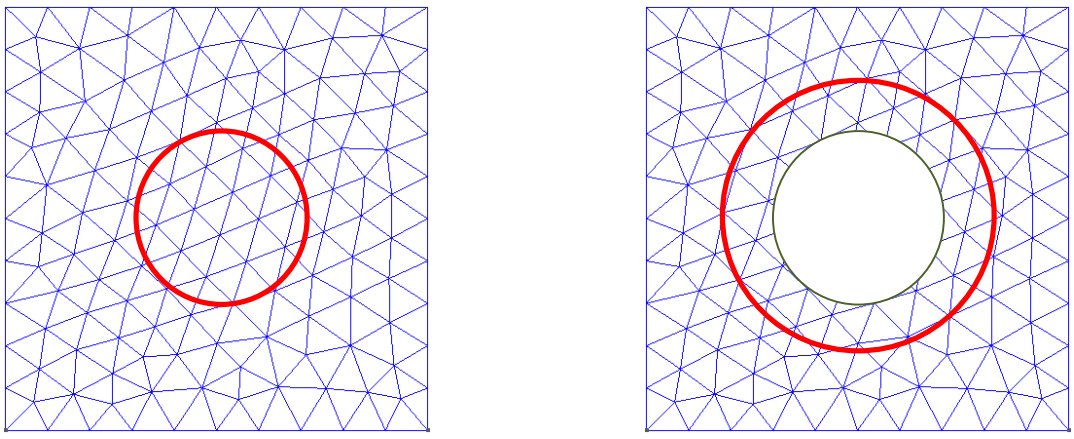


Figure 4.5. *Circular hole.*

$$err_{ext} = \left(0.7\left[\left(\frac{\theta}{2} - \frac{\sin 2\theta}{4} \right)\Big|^{2\pi} - \left(\frac{\theta}{2} - \frac{\sin 2\theta}{4} \right)\Big|^{0} \right] \right)^{1/2} = \left(0.7\left[\pi - \frac{\sin 4\pi}{4} + \frac{\sin 0}{4} \right] \right)^{1/2}$$

$$err_{ext} = \left(r\int_0^{2\pi} \sin^2\theta d\theta \right)^{1/2} = \left(0.4\left[\pi - \frac{\sin 4\pi}{4} + \frac{\sin 0}{4} \right] \right)^{1/2} = 1.482941286$$

On the below meshes, we show some of the meshes used for the computation on the plate. These meshes and all the computations have been obtained with the C++ extended finite element methods tools which developed by Ecole Centrale de Nantes team.

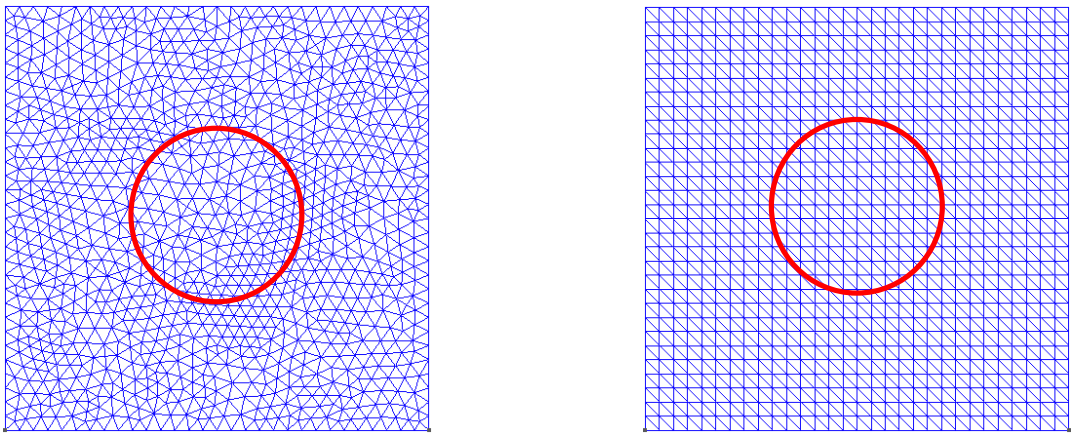


(a) (b)

Figure 4.6. *One of the computation with (10 x 10 elements): (a) in example 1; (b) in example 2.*

A couple of computations with different meshes have been made. Six different meshes are usually considered in each type of problems: 5 x 5 (0.400), 10 x 10 (0.200), 20 x 20 (0.100), 40 x 40 (0.050), 80 x 80 (0.025), 160 x 160 (0.0125). Furthermore, we also use the other meshes represented in tables. A totally unstructured mesh could also have been used. A sample unstructured mesh (10 x 10 elements) is shown in Figure 4.6.

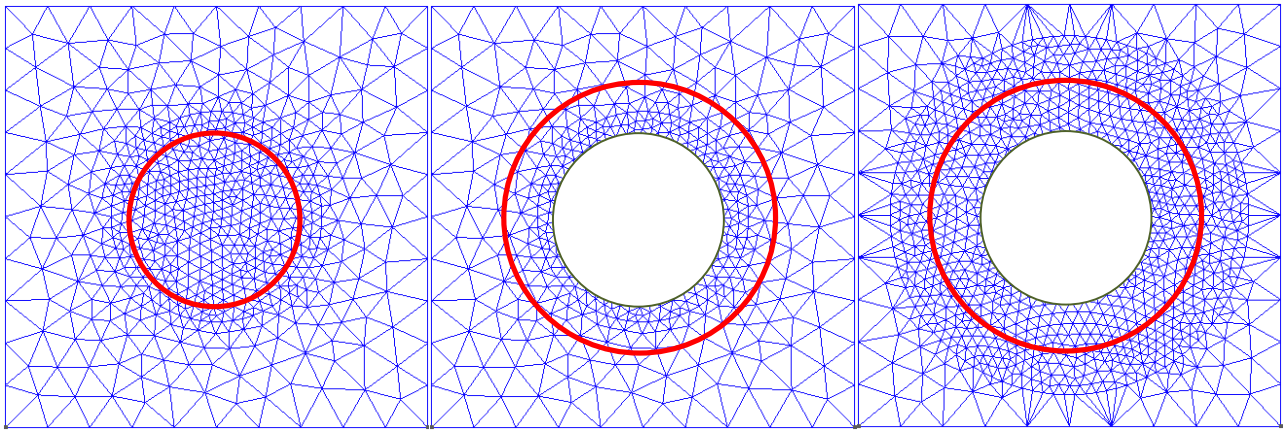
Moreover, we use the computations with unstructured meshes and structured meshes. The sample unstructured mesh and structured mesh (30 x 30 elements) is shown in Figure 4.7.



(a) (b)

Figure 4.7. *Examples of mesh (30x30 elements) used for the plate: (a) unstructured mesh (2457 dofs); (b) structured mesh (2183 dofs).*

These mesh is obtained by meshing a plate with a hole of radius R=0.4 and by meshing also the interior of this hole. The computational domain and the unstructured mesh are shown in Figure 4.8.



(a) (b) (c)

Figure 4.8. *For unstructured mesh (10x10 elements): (a) refined mesh with the radius equal to 0.5 for example 1 (1531 dofs); (b) refined mesh with the radius equal to 0.5 for example 2 (1133 dofs); (c) refined mesh with the radius equal to 0.85 for example 2 (2649 dofs).*

4.1 Example 1

Table 4.1: *Unstructured mesh computations on the first example.*

Mesh	5x5	10x10	20x20	40x40	50x50	60x60	70x70	80x80	160x160
energy exact	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64
error absolute exact	0.129892	0.076799	0.020652	0.007106	0.005427	0.003529	0.001986	0.001687	0.000776
error relative exact	0.068082	0.040254	0.010825	0.003725	0.002844	0.00185	0.001041	0.000884	0.000407
energy exact on LM	1.01701	1.24049	1.24732	1.25355	1.25729	1.25412	1.25525	1.25516	1.25615
error absolute exact on LM	0.417561	0.194305	0.093589	0.044254	0.039844	0.031056	0.023019	0.021145	0.011782
error relative exact on LM	0.414054	0.174457	0.083799	0.039526	0.035534	0.027732	0.020546	0.018874	0.010512
Inf-Sup	1.203841	0.84819	0.670264	0.574195	*	*	*	*	*

Table 4.2: *Structured mesh computations on the first example.*

Mesh size	5x5	10x10	20x20	30x30	40x40	80x80	160x160
energy exact	3.64	3.64	3.64	3.64	3.64	3.64	3.64
error absolute exact	0.135507	0.074203	0.016705	0.007541	0.004913	0.001321	0.000573
error relative exact	0.071025	0.038893	0.008756	0.003953	0.002575	0.000693	0.0003
energy exact on LM	1.1064	1.20974	1.24393	1.24677	1.24962	1.256	1.25588
error absolute exact on LM	0.463372	0.20336	0.099764	0.066784	0.050617	0.023894	0.012239
error relative exact on LM	0.440528	0.184892	0.089449	0.059811	0.04528	0.021321	0.010922
Inf-Sup	0.883513	0.633505	0.539896	0.539956	0.535193	*	*

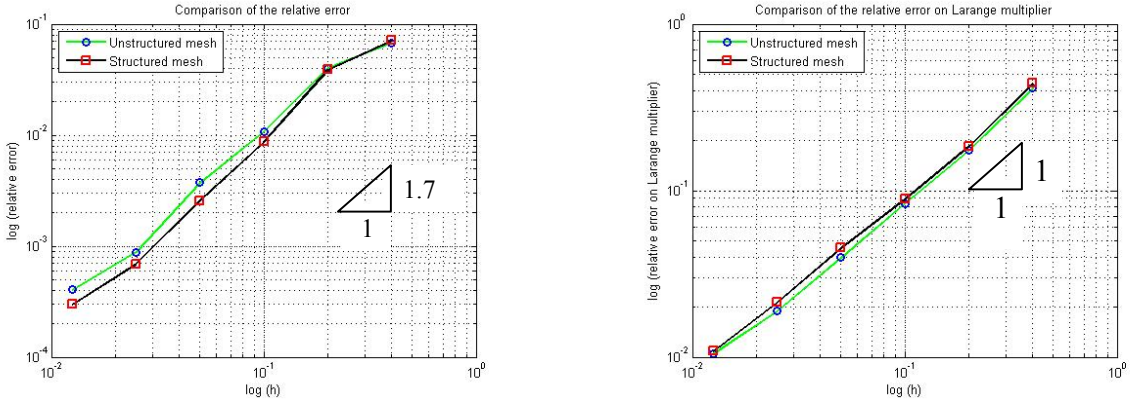
Table 4.3: *Unstructured refined mesh (with radius = 0.5) computations on the first example.*

Mesh	5x5	7x7	9x9	10x10	11x11	12x12	13x13	14x14	15x15	20x20	25x25	30x30
energy exact	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64	3.64
error absolute exact	0.025119	0.009058	0.006048	0.005923	0.005114	0.004625	0.003775	0.00309	0.002399	0.00207	0.001476	0.000946
error relative exact	0.013166	0.004747	0.00317	0.003104	0.00268	0.002424	0.001979	0.00162	0.001258	0.001085	0.000774	0.000496
energy exact on LM	1.2786	1.25702	1.25648	1.26358	1.25878	1.25372	1.25664	1.26122	1.25659	1.25726	1.25577	1.25713
error absolute exact on LM	0.110333	0.061094	0.047165	0.045261	0.040933	0.038293	0.032205	0.031895	0.029489	0.021917	0.018099	0.013605
error relative exact on LM	0.097575	0.054491	0.042077	0.040265	0.036484	0.034199	0.028729	0.0284	0.026307	0.019547	0.016151	0.012134
Inf-Sup	0.430819	0.355826	0.349526	0.330878	0.3321	0.341292	0.317271	0.32176	*	*	*	*

Remark: * shown that we are not able to obtain the computation of inf-sup value. That is a problem because the number of dofs is "high". The capability of the computer was a limit for those computation. In those limited computations, it was not possible to use a mesh as fine as we wanted to obtain inf-sup value.

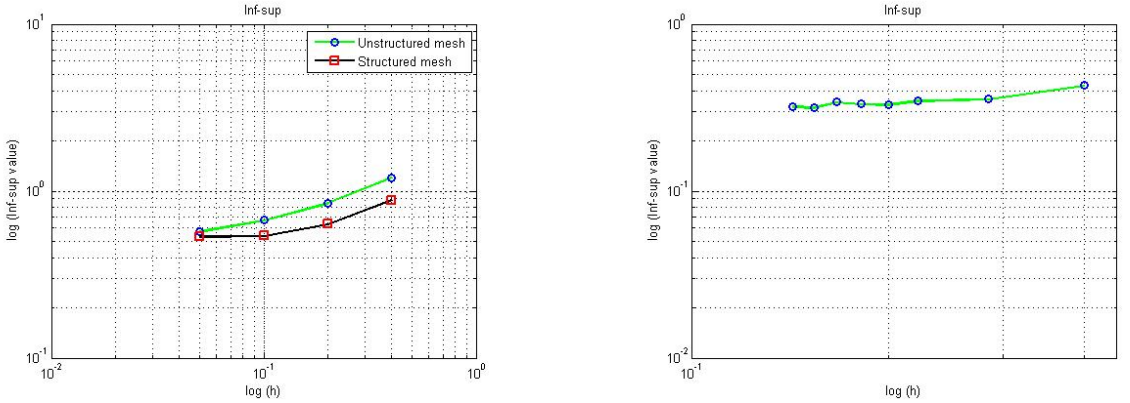
We test the X-FEM on a problem to establish the rate of convergence of the method. The convergence of the numerical method is studied by the energy norm and the Lagrange multiplier. The relative error in the energy norm is computed as the mesh is refined. The relative error in the energy norm is plotted as a function of the mesh spacing (log-log plot). In theory the plot of the energy in function of the number of dof. in scale log-log is straight line if there is convergence and the slope of it (<1) give the rate of convergence.

For measuring the accuracy of the computation, in these figures all of them converge with mesh refinement. The energy convergence will be used. These computations will verify that we have a convergence for all test cases.



(a) (b)

Figure 4.9. *For the example 1, the comparison between structured and unstructured meshes: (a) convergence of the energy norm; (b) convergence of the* L^2 *error on the Lagrange multiplier.*



(a) (b)

Figure 4.10. *For the example 1: (a) inf-sup value on the comparison between structured and unstructured meshes; (b) inf-sup value of the refined mesh.*

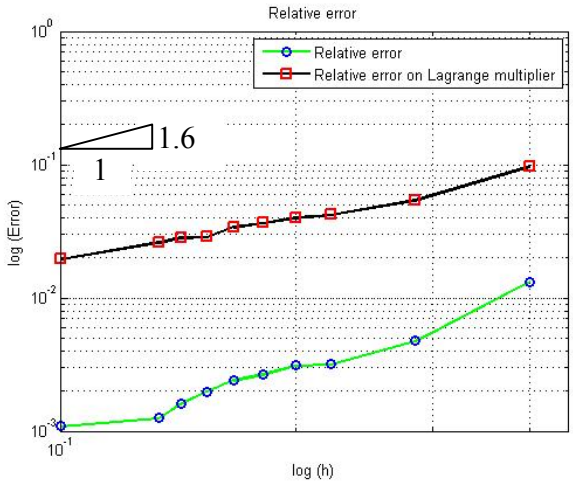
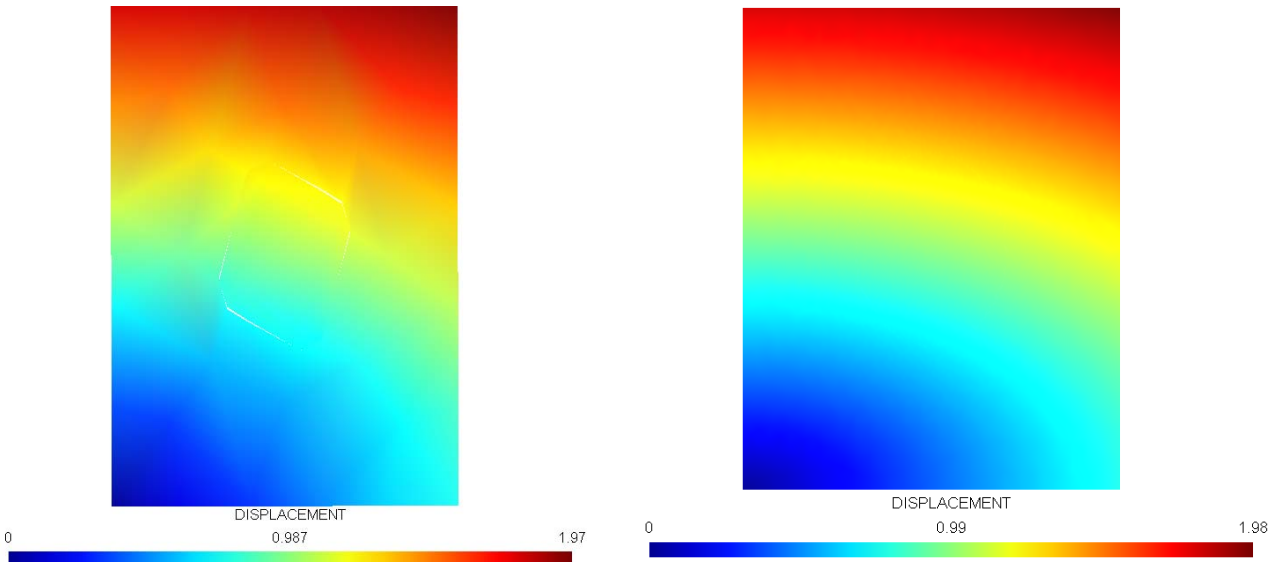


Figure 4.11. *Error on the Lagrange multipliers for the interface problem (refined mesh at the radius equal to 0.5).*

In order to investigate the behavior of the new approach, we computed the L^2 norm for each simulation. Results for the non-matching (uniform) meshes case using of the reduced multiplier space: convergence of the errors and inf-sup parameter.

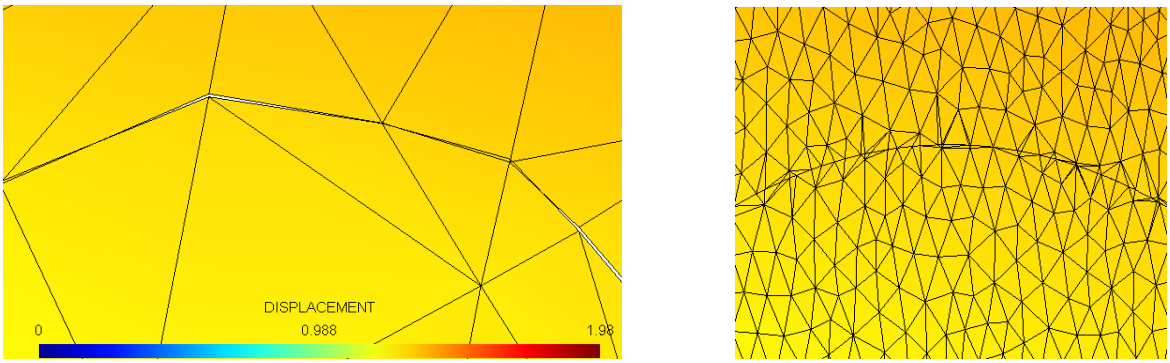
Fields of stress and displacement are computed by the extended finite element method in square plate. Figures 4.12 and 4.14 shows a comparison among displacement and stress fields.



(a) (b)

Figure 4.12. *For example 1, the comparisons between displacements in: (a) mesh (5x5 elements, 103 dofs); (b) mesh(160x160 elements, 51999 dofs).*

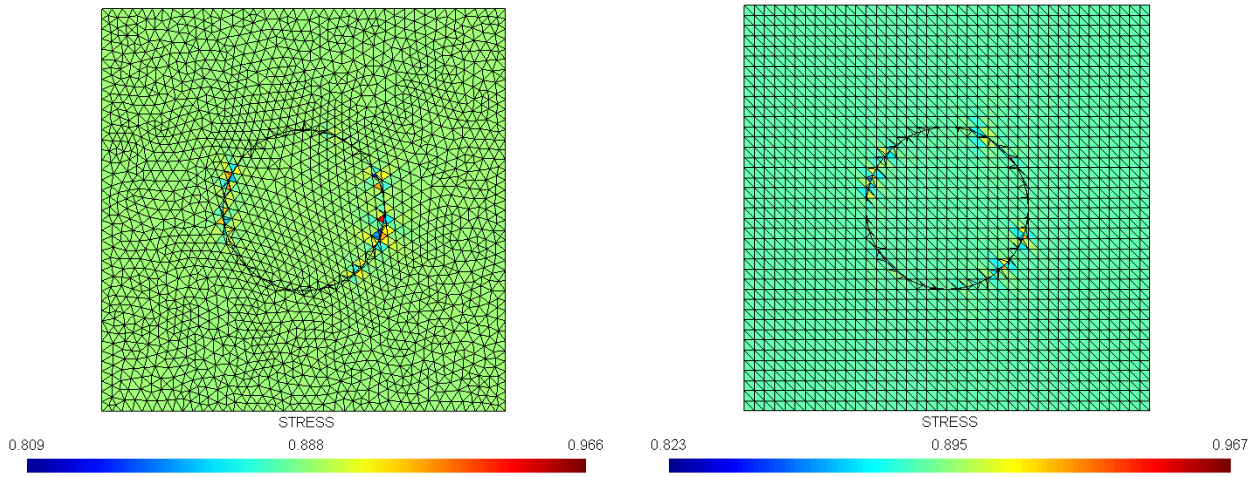
Please note the gap (jump in the displacements) in a rough mesh and a fine mesh. In the other comparison, make both mesh visible and we'll have what looks like on the appear mesh again. No improvement is obtained in the coarse mesh. However, the result is good for the fined mesh.



(a) (b)

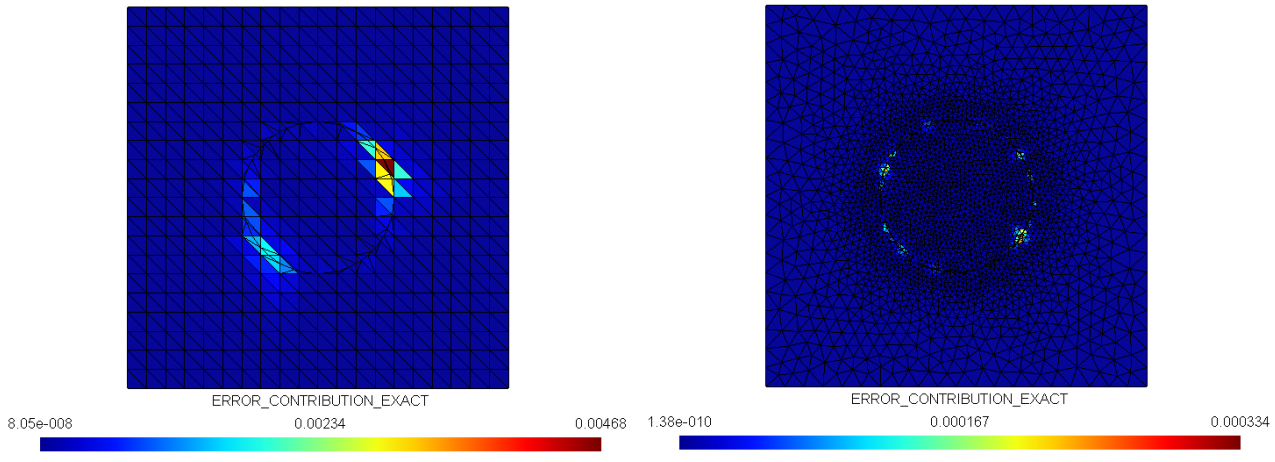
Figure 4.13. *Zoom on the interface of example 1 for mesh: (a) (10x10 elements, 341 dofs); (b) (160x160 elements, 51999 dofs).*

As the study relative to the displacement fields leads to good conclusions, we now focus on stresses within the plate.



(a) (b)

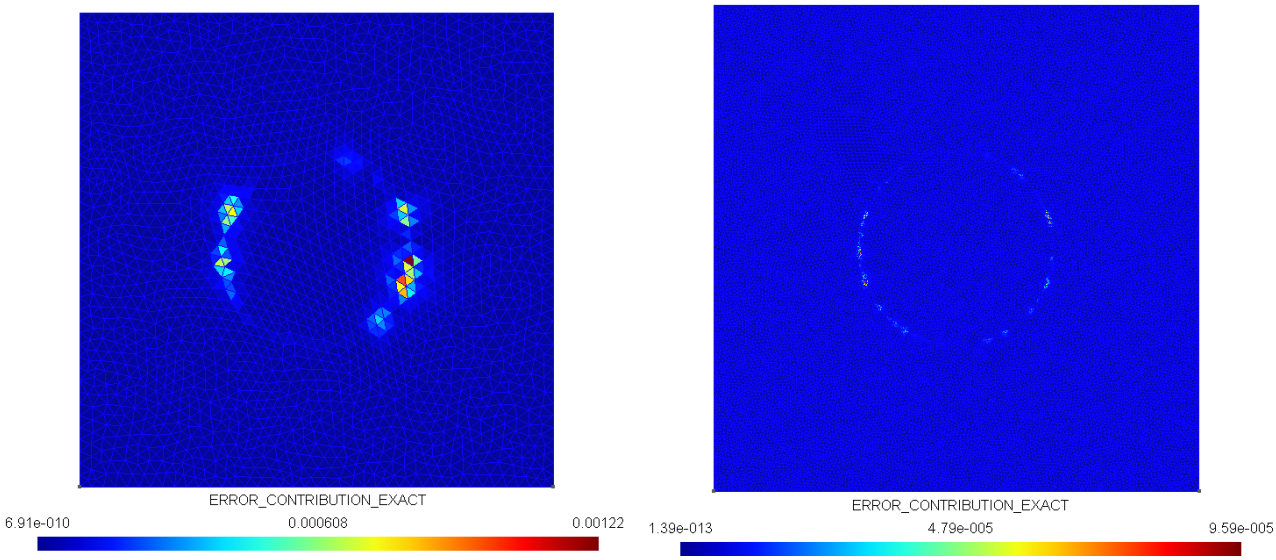
Figure 4.14. *Comparison of stress in the mesh (40x40 elements): (a) unstructured mesh (4225 dofs); (b) structured mesh (3661 dofs).*



(a) (b)

Figure 4.15. *Comparison of the mesh (20x20 elements): (a) structured mesh (1029 dofs); (b) unstructured mesh with the refined mesh in the radius equal to 0.5 (5717 dofs).*

And for computing error,



(a) (b)

Figure 4.16. *The error computations are carried out for different values of mesh: (a) mesh of 40x40 elements (4225 dofs); (b) mesh of 160x160 elements (51999 dofs).*

4.2 Example 2

Table 4.4: *Unstructured mesh computations on the second example.*

Mesh	5x5	10x10	20x20	25x25	30x30	35x35	40x40	50x50	80x80	160x160
energy exact	4.529890	4.062010	4.032410	4.029740	4.026630	4.026170	4.025730	4.024650	4.023090	4.022620
error absolute exact	0.834429	0.328234	0.212376	0.163092	0.126669	0.113101	0.101604	0.085112	0.053672	0.025131
error relative exact	0.392054	0.162859	0.105760	0.081245	0.063125	0.056366	0.050639	0.042426	0.026759	0.012530
energy exact on LM	1.451080	1.461360	1.480620	1.479230	1.476740	1.476130	1.477590	1.475510	1.477540	1.475550
error absolute exact on LM	0.530019	0.179287	0.103852	0.078560	0.063427	0.053488	0.045613	0.036837	0.021957	0.010921
error relative exact on LM	0.439992	0.148310	0.085348	0.064593	0.052194	0.044024	0.037524	0.030326	0.018063	0.008990
Inf-Sup	1.067191	0.691359	0.650401	0.625104	0.667652	0.683250	0.728837	*	*	*

Table 4.5: *Structured mesh computations on the second example.*

Mesh	5x5	10x10	20x20	30x30	35x35	40x40
energy exact	4.492590	4.082770	4.036700	4.028810	4.027060	4.026090
error absolute exact	0.767359	0.379316	0.223537	0.160995	0.145170	0.126327
error relative exact	0.362034	0.187726	0.111259	0.080209	0.072341	0.062958
energy exact on LM	1.367060	1.536730	1.495210	1.484430	1.471700	1.479790
error absolute exact on LM	0.546072	0.296083	0.125341	0.069991	0.063593	0.056134
error relative exact on LM	0.467042	0.238844	0.102504	0.057446	0.052420	0.046145
Inf-Sup	0.691551	0.621886	0.587801	0.546792	0.531919	0.554174

Table 4.6: *Unstructured refined mesh (with radius = 0.5) computations on the second example.*

Mesh	5x5	10x10	15x15	20x20	25x25	30x30
energy exact	4.036110	4.025520	4.023750	4.023280	4.022920	4.022810
error absolute exact	0.308659	0.148416	0.099044	0.074615	0.060733	0.050027
error relative exact	0.153638	0.073973	0.049376	0.037199	0.030280	0.024942
energy exact on LM	1.462390	1.490250	1.480360	1.480510	1.477460	1.477000
error absolute exact on LM	0.199182	0.100781	0.073558	0.046938	0.039499	0.031136
error relative exact on LM	0.164709	0.082556	0.060457	0.038576	0.032496	0.025620
Inf-Sup	0.530101	0.490511	0.511335	*	*	*

Table 4.7: *Unstructured refined mesh (with radius = 0.85) computations on the second example.*

Mesh	3x3	5x5	6x6	8x8	9x9	10x10	11x11	12x12	14x14	15x15	20x20
energy exact	4.055570	4.033510	4.029880	4.028180	4.025700	4.024840	4.024610	4.024470	4.024180	4.023800	4.023120
error absolute exact	0.341845	0.217638	0.178971	0.149987	0.127157	0.110835	0.104786	0.099668	0.087163	0.077969	0.058261
error relative exact	0.169748	0.108366	0.089153	0.074731	0.063375	0.055246	0.052232	0.049682	0.043450	0.038869	0.029047
energy exact on LM	1.518900	1.474600	1.481080	1.472280	1.477390	1.475340	1.476100	1.476800	1.478030	1.475990	1.476870
error absolute exact on LM	0.207665	0.104269	0.086510	0.057385	0.054747	0.047841	0.044738	0.040658	0.030778	0.028799	0.022523
error relative exact on LM	0.168500	0.085866	0.071085	0.047293	0.045042	0.039387	0.036823	0.033457	0.025316	0.023705	0.018533
Inf-Sup	0.399471	0.354536	0.365466	0.334062	0.341433	0.344741	0.329230	*	*	*	*

The results in this section are presented for a plate with a hole of 0.4 of radius. The inf-sup parameter (numerical inf-sup test) is given as well as the convergence of the errors. The upper of the four figures are the energy error, the following two figures are the error on the Lagrange multipliers and the last two bottom figures is the inf-sup value.

Rate of convergence in energy for the plate with a hole problem. This slopes obtained in figures proved that the code do converge in energy.

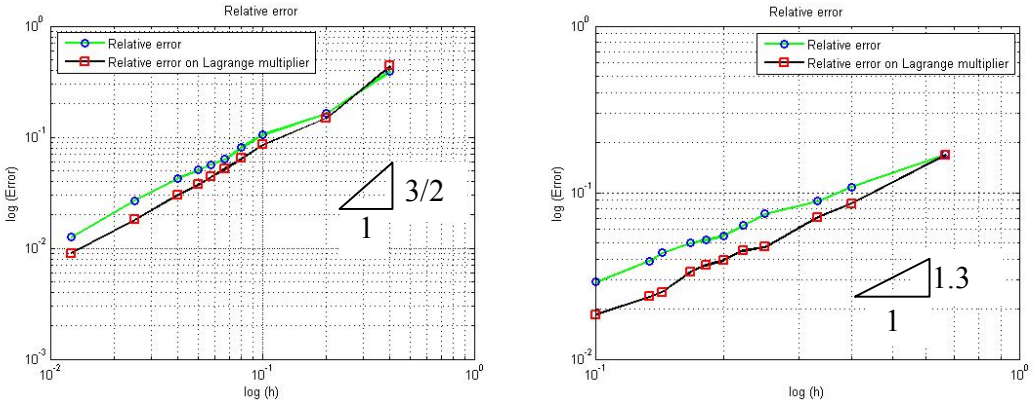
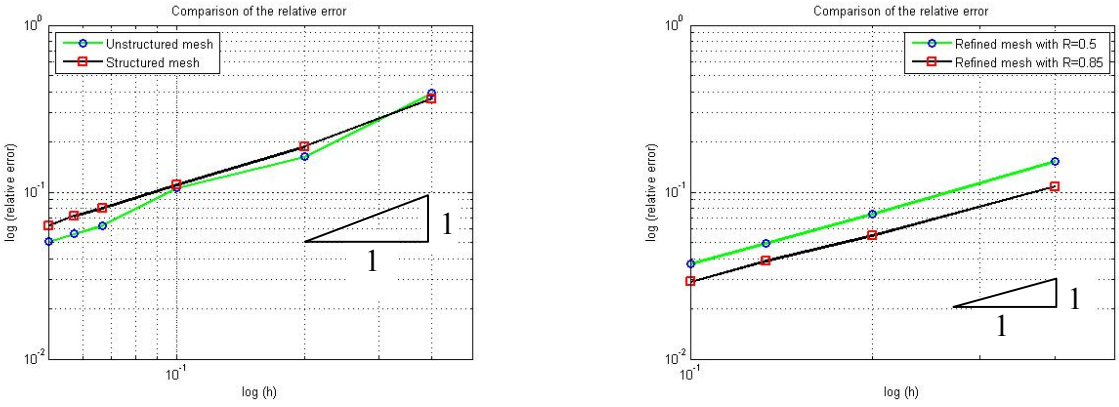


Figure 4.17. *Error evolution for non-matching meshes.*

Fig 4.17 shows a convergence of the path with mesh refinement. Also the path has smaller oscillations with mesh refinement. This is due to the convergence for fined meshes.

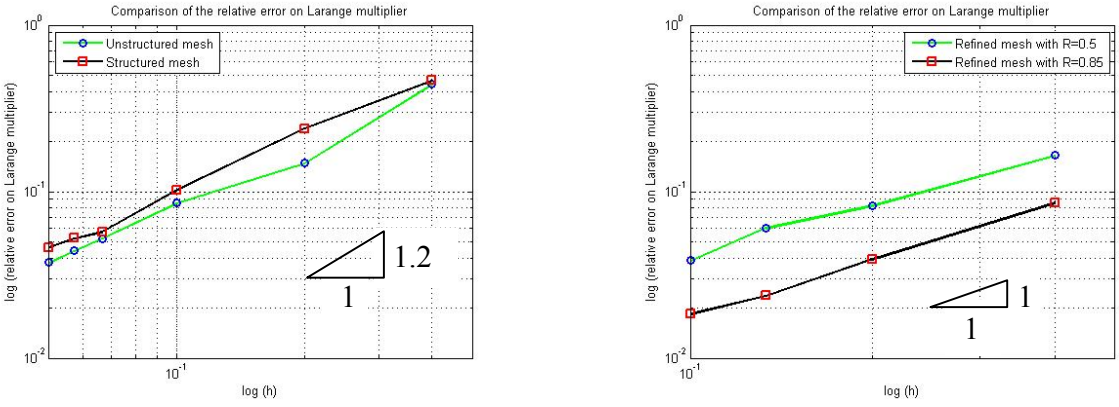


(a) (b)

Figure 4.18. *Convergence of the relative error: (a) without refined mesh; (b)with circular refined mesh.*

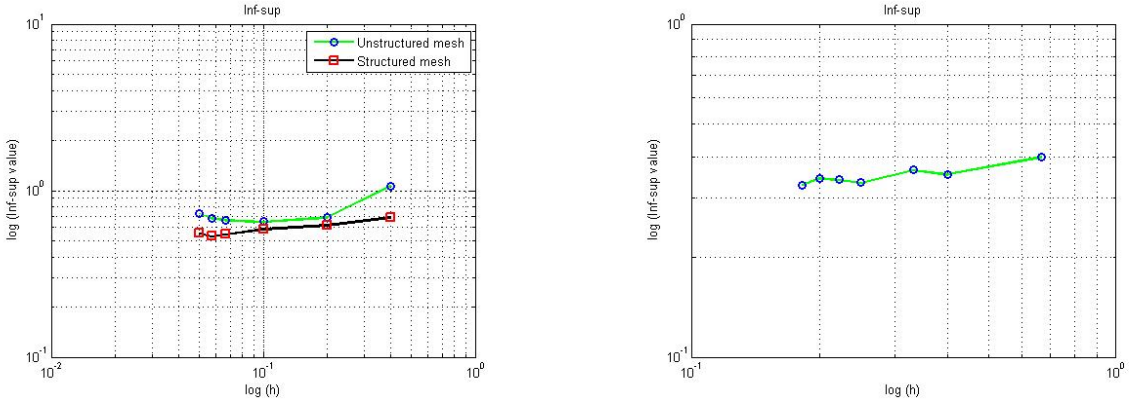
In Figure 4.18 this is clearly shown, it seems that for refined meshes the rate of convergence becomes more stable.

The relative error norms on the Lagrange multipliers are shown.



(a) (b)

Figure 4.19. *Error on the Lagrange multipliers for the interface problem: (a) unstructured and structured meshes; (b) refined mesh at the radius equal to 0.5 and 0.85.*

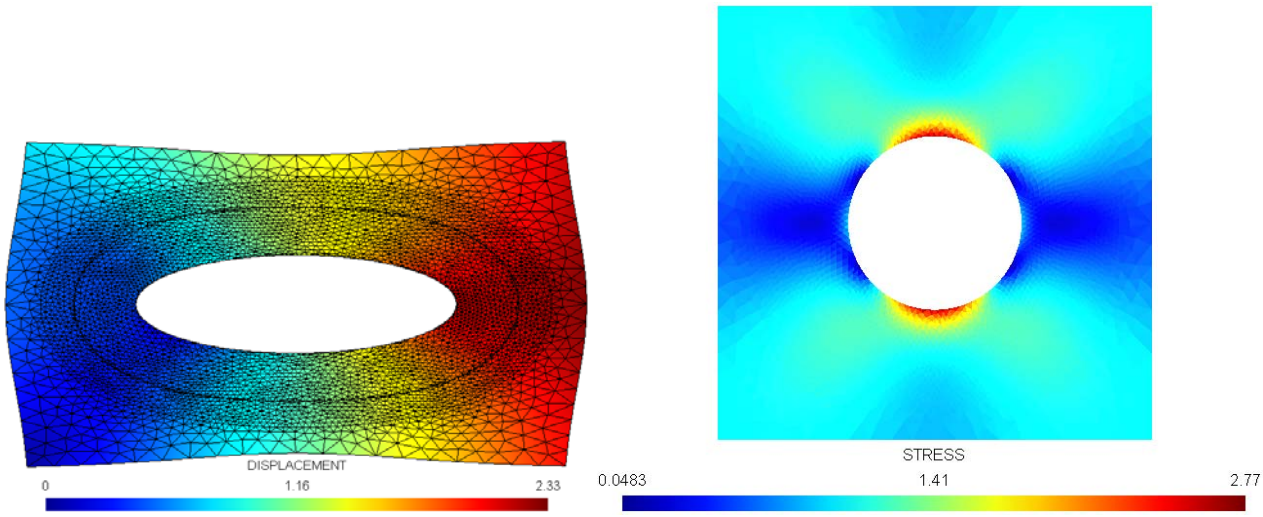


(a) (b)

Figure 4.20. *Numerical computed inf-sup value: (a) without refined radius; (b) refine with the radius equal to 0.85.*

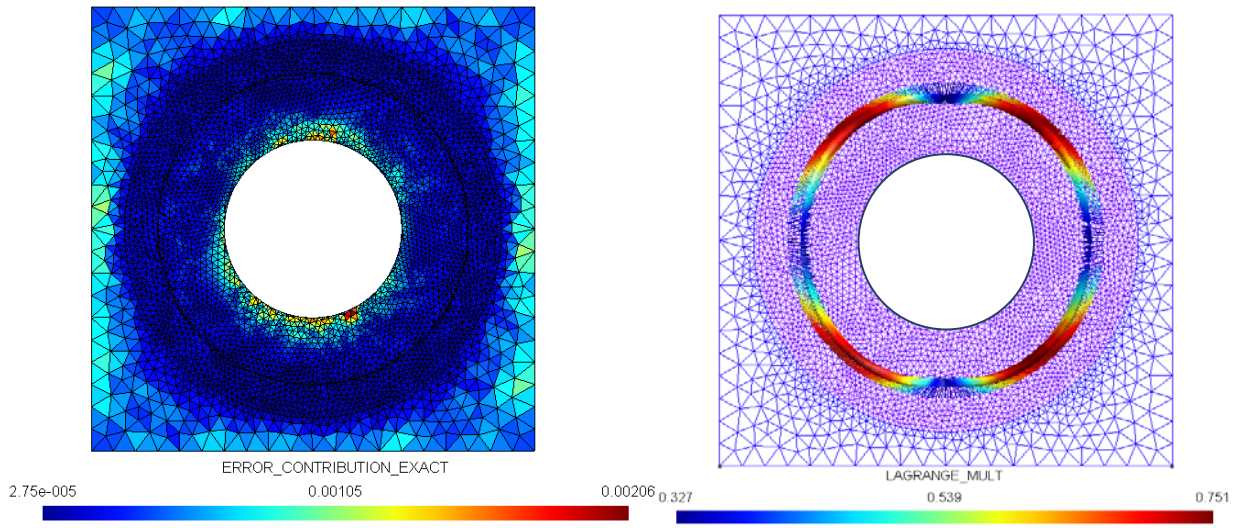
In Figure 4.19, the convergence rates are indicated on the plots. In Figure 4.20, the inf-sup conditions do not tend to zero if the mesh-size tends to zero.

These computation have been made with (20x20 elements).



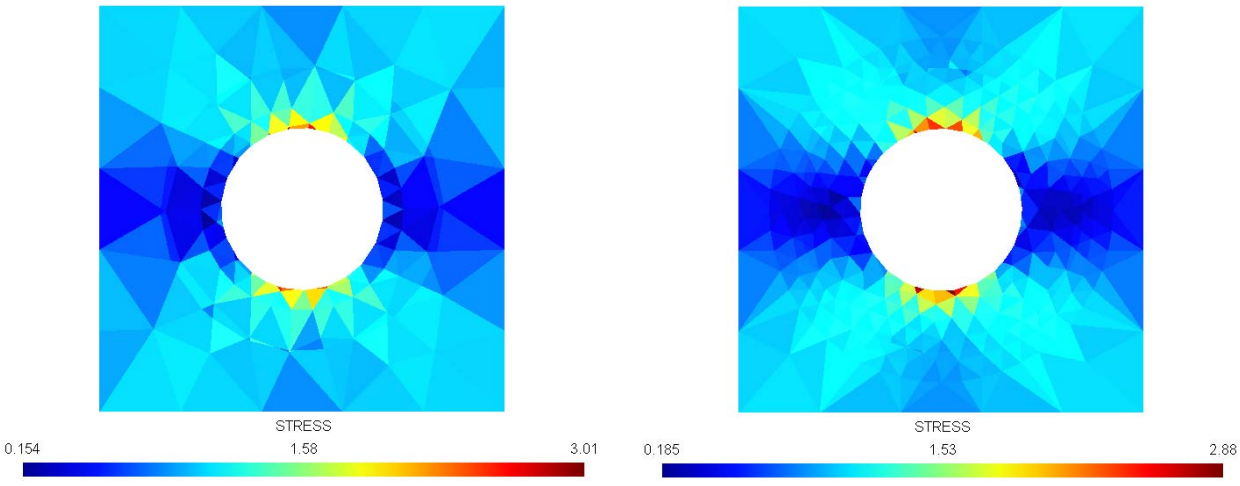
(a) (b)

Figure 4.21. *Mesh (20x20 elements, 9231 dofs): (a) the displacement field; (b) stress field in the body.*



(a) (b)

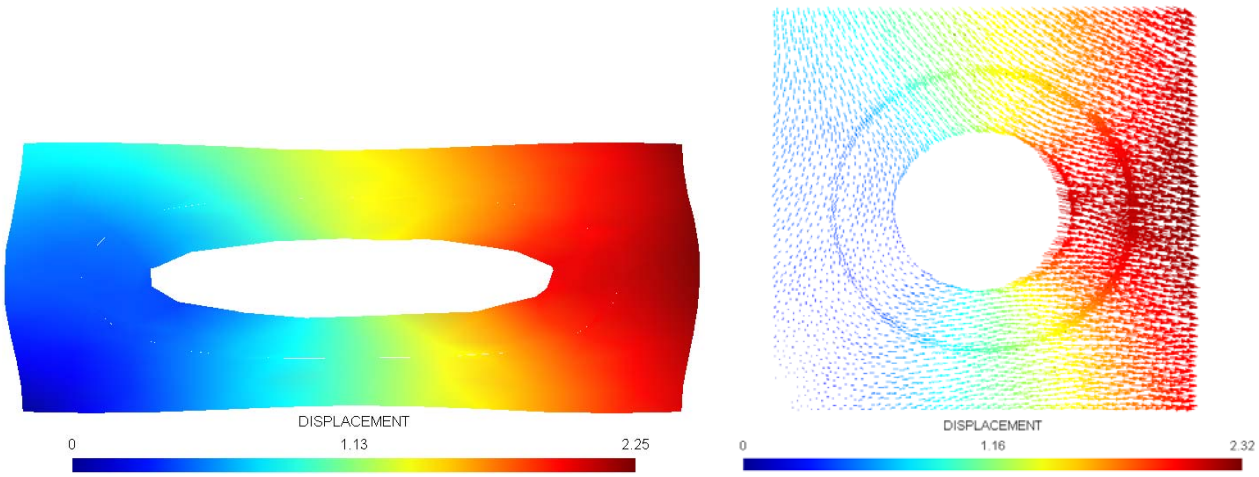
Figure 4.22. *Mesh (20x20 elements, 9231 dofs): (a) error; (b) vectors of Lagrange multipliers.*



(a) (b)

Figure 4.23. *For example 2, in the mesh (10x10 elements) the comparison between stresses in: (a) the refined mesh with the radius equal to 0.5 (1133 dofs); (b) the refined mesh with the radius equal to 0.85 (2649 dofs).*

The deformation obtained for this reference solution is represented on Figure 4.24.



(a) (b)

Figure 4.24. *The displacement field in the plate with: (a) a mesh (10x10 elements, 417 dofs) display as continuous map; (b) a mesh (40x40 elements, 2623 dofs) display as vector field.*

In the numerical computations, it is shown that the stability issue should be clarified and properly analyzed. All these results came from computations with X-FEM. The error in energy norm, the error on the Lagrange multipliers and the inf-sup are used to evaluate the resulting algorithm. We described the error as a function of the mesh size. We prove energy type in 2 space dimensions, and present some numerical examples. For two cases all computations have been made to compare the results. In this study, the results obtained for the test case seems logical.

Chapter 5
Conclusions

The aim of this thesis is to present a general overview on the existing techniques to enforce essential boundary conditions in X-FEM. From the general to the particular, the application of the X-FEM to model the kinematic condition in two-dimensional solid mechanics is explored. It is shown that, the Lagrange multiplier method is one of the most popular, because of its straightforward implementation and applicability to a large variety of problems. However, attention must be paid to the choice of the interpolation space for the Lagrange multiplier. The discretization of the Lagrange multiplier must be accurate enough in order to obtain an acceptable solution, but it can lead to singular matrices if the interpolation space does not verify the Ladygenskaya-Babuška-Brezzi stability condition. Thus, the stability issue should be clarified and properly analyzed.

A simple 2D linear elasticity problem shows the major difficulties in the practical choice of the interpolation of the multiplier in particular situations. Particular examples are used to analyze and compare their performance in different situations. The computational time and costs incurred in the implementation and execution of any numerical method is a critical component in the evaluation of its feasibility, usability, and potential for applications. The numerical results demonstrated the capabilities, versatility, accuracy, and robustness of the numerical method.

References

[1] A. Nouy, F. Schoefs and N. Moës, “X-SFEM a computational technique based on X-FEM to deal with random shapes”, *European Journal of Computational Mechanics*, Vol. 16, 277–293, 2007.

[2] Bathe K.J., “Finite Element Procedures in Engineering Analysis”, 2nd edition, *Prentice-Hall*, Englewood Cliffs, New Jersey, 1996.

[3] Bathe K.J., “The inf-sup condition and its evaluation for mixed finite element methods”, *Computers & Structures*, Volume 79, Issue 9, 243–252, 971, April 2001.

[4] Bathe K.J., Chapelle D., “The inf-sup test”, *Computers and Structures*, Vol. 47, pp. 537-545, 1993.

[5] Bathe K.J., Iosilevich, A., Chapelle, D., “An inf-sup test for shell finite elements”, *Computers and Structures*, Vol. 75, pp. 439-456, 2000.

[6] C. Daux, N. Moës, J. Dolbow, N. Sukumar and T. Belytschko, “Arbitrary branched and Intersecting Cracks with the eXtended Finite Element Method”, *International Journal for Numerical Methods in Engineering*, Vol. 48, Number 12, pp. 1741–1760, 2000.

[7] E. Béchet, H. Minnebo, N. Moës, B. Burgardt, “Improved implementation and robustness study of the X-FEM for stress analysis around cracks”, *International Journal for Numerical Methods in Engineering*, 64, 1033–1056, 2005.

[8] E. Béchet, N. Moës, B. Wohlmuth, “A stable Lagrange multiplier space for the stiff interface conditions within the extended finite element method”, *International Journal for Numerical Methods in Engineering*, 2008.

[9] F. Shao, K. V. Ling and W. S. Ng, ”3D Prostate Surface Detection from Ultrasound Images Based on Level Set Method”, *Springer*, Dohi and R. Kikinis (Eds): LNCS 2489, Tokyo, Japan, pp. 389-396, 2002.

[10] G. Legrain, N. Moës, A. Huerta, “Stability of incompressible formulations enriched with X-FEM”, *Comp. Methods Appl. Mech. Engrg*., 197, 1835–1849, 2008.

[11] G. Legrain, N. Moës, E. Verron, “Stress analysis around crack tips in finite strain problems using the eXtended finite element method”, *International Journal for Numerical Methods in Engineering*, 63, 290–314, 2005.

[12] Hansbo A., Hansbo P. and Larson M.G., “A finite element method on composite grids based on Nitche's method”, *ESAIM: M2AN*, v37. 495-514, 2003.

[13] Hansbo, P., C. Lovadina, I. Perugia, G. Sangalli, “A Lagrange multiplier method for the finite element solution of elliptic interface problems using non-matching meshes”, *Numer. Math*., Vol. 100 (1), pp. 91-115, 2005.

[14] H. Ji, Introduction to the X-FEM and comparison of approaches used to satisfy the interface conditions, Duke University, 2001.

[15] H. Ji, J.E. Dolbow, “On strategies for enforcing interfacial constraints and evaluating jump conditions with the extended finite element method”, *International Journal for Numerical Methods in Engineering*, 61, 2508–2535, 2004.

[16] I. Babuška, “The finite element method with Lagrangian multipliers”, *Numerical Math*., 20, 179-192, 1973.

[17] I. Babuška, “The finite element method with penalty”, *Math. Comp.*, 27, 221-228, 1973.

[18] I. Babuška, J.M. Melenk, “The partition of unity method”, *International Journal for Numerical Methods in Engineering*, 40, 727–758, 1997.

[19] I. Nistor, M. L. E. Guiton, P. Massin, N. Moës, S. Géniaut, “An X-FEM approach for large sliding contact along discontinuities”, *Wiley*, 2008.

[20] Jasjit S. Suri, David L. Wilson and Swamy Laxminarayan, “Handbook of biomedical image analysis: Volume I: Segmentation models Part A”, *Kluwer Academic / Plenum Publishers*, New York, 2005.

[21] J. Dolbow, N. Moës, T. Belytschko, “Discontinuous enrichment in finite elements with a partition of unity method”, *Finite Elem. Anal. Des*., 36, 235–260, 2000.

[22] J. Dolbow, N. Moës, T. Belytschko, “Modeling fracture in Mindlin-Reissner plates with the extended finite element method”, *Int. J. Solids Struct.*, vol. 37, pp 7161-7183, 2000.

[23] J. Dolbow, N. Moës, T. Belytschko, “An extended finite element method for modeling crack growth with frictional contact”, *Comp. Methods Appl. Mech. Engrg.*, 190, 6825–6846, 2001.

[24] J. M. Melenk and I. Babuska, “The partition of unity method: basic theory and applications”, *Computer Methods in Applied Mechanics and Engineering*, Vol. 39, 289–314, 1996.

[25] J. Nitsche, “Über ein Variationsprinzip zur Losung von Dirichlet-Problemen bei Vervendung von Teilraumen, die keinen Randbedingungen unterworfen sind”, *Abhandlungen aus dem Mathematischen Seminaren des Universitat Hamburg, 36*, 9-15, 1970-1971.

[26] Matthieu Tourbier, "Contribution à la résolution de problèmes d’interfaces mobiles de type Dirichlet", Phd thesis, *Ecole Centrale de Nantes et Université de Nantes*, 2005.

[27] Nagi El-Abbasi and Klaus-Jürgen Bathe, “Stability and patch test performance of contact discretizations and a new solution algorithm”, *Computers & Structures*, Volume 79, Issue 16, Pages 1473-1486, June 2001.

[28] N. Moës, “The eXtended Finite Element Method (X-FEM)”, class notes, *Ecole Centrale de Nantes*.

[29] N. Moës, E. Béchet, M. Tourbier, “Imposing Dirichlet boundary conditions in the extended finite element method”, *International Journal for Numerical Methods in Engineering*, 67, 1641–1669, 2006.

[30] N. Moës, J. Dolbow, T. Belytschko, “A finite element method for crack growth without remeshing”, *International Journal for Numerical Methods in Engineering*, vol. 46, pp 131-150, 1999.

[31] N. Moës, N. Sukumar, B. Moran and T. Belytschko, “An eXtended Finite Element Method (X-FEM) for Two- and Three-Dimensional Crack Modeling”, in *ECCOMAS 2000*, Barcelona, Spain, September 11–14, 2000.

[32] N. Moës, M. Cloirec, P. Cartraud and J.-F. Remacle, “A computational approach to handle complex microstructure geometries”, *Comp. Meth. Appl. Mech. Eng.*, Vol. 192, 3163–3177, 2003.

[33] N. Moës, T. Belytschko, “Extended finite element method for cohesive crack growth”, *Eng. Fract. Mech.*, 69, 813–833, 2002.

[34] N. Sukumar, D. L. Chopp, E. Béchet and N. Moës, “Three-Dimensional Non-Planar Crack Growth by a Coupled eXtended Finite Element and Fast Marching Method”, *International Journal for Numerical Methods in Engineering*, Vol. 76, Number 5, pp. 727–748, 2008.

[35] N. Sukumar, D. L. Chopp, N. Moës and T. Belytschko, “Modeling Holes and Inclusions by Level Sets in the eXtended Finite Element Method”, *Computer Methods in Applied Mechanics and Engineering*, Vol. 190, Number 46–47, pp. 6183–6200, 2001.

[36] N. Sukumar, N. Moës, B. Moran and T. Belytschko, “Extended Finite Element Method for Three-Dimensional Crack Modelling”, *International Journal for Numerical Methods in Engineering*, Vol. 48, Number 11, pp. 1549–1570, 2000.

[37] Patrick Hild, Yves Renard, "A stabilized Lagrange multiplier method for the finite element approximation of contact problems in elastostatics", *Prépublications du laboratoire de mathématiques de Basançon*, n° 20, 2007.

[38] Patrick Massin, Samuel Geniaut, N. Moës, "Fissuration avec X-FEM et contact", *Giens*, 2005.

[39] Per Heintz, Peter Hansbo, “Stabilized Lagrange multiplier methods for bilateral elastic contact with friction”, *Comp. Methods Appl. Mech. Engrg.*, Volume 195, Issues 33-36, 4323-4333, 2006.

[40] P. Lancaster, K. Salkauskas, “Surfaces generated by moving least-squares methods”, *Mathematics and Computations*, Vol. 37, No. 155, pp. 141-158, 1981.

[41] P. Rozycki, N. Moës, E. Béchet, C. Dubois, “X-FEM explicit dynamics for constant strain elements to alleviate mesh constraints on internal or external boundaries", *Comp. Methods Appl. Mech. Engrg.*, 197, 349–363, 2008.

[42] P. Wriggers, “Computational Contact Mechanics”, *Wiley*, 2002.

[43] Samuel Geniaut, "Approche X-FEM Pour La Fissuration Sous Contact Des Structures Industrielles", Phd thesis, *Ecole Centrale de Nantes et Université de Nantes*, 2006.

[44] Solberg, J. M. and Papadopoulos, P., “An analysis of dual formulations for the finite element solution of two-body contact problems”, *Computer Methods in Applied Mechanics and Engineering*, 194 (25–26): 2734–2780, 2005.

[45] Stéphane BORDAS, “Extended Finite Element and Level Set Methods with Applications to Growth of Cracks and Biofilms”, Phd thesis, *Northwestern University*, 2003.

[46] S. Bertoluzza, F. Brezzi, G. Sangalli, “The method of mothers for non-overlapping non-matching DDM”, *Numer. Math.*, Vol. 107 (3), pp. 397-431, 2007.

[47] S. Bertoluzza, F. Brezzi, L.D. Marini, G. Sangalli, “Non-matching Grids and Lagrange Multipliers”, (Proceedings of *DDM 15* in: Domain Decomposition Methods in Science and Engineering, Berlin, Germany, July 21-25, 2003).

[48] S. De, "Generalized Finite Element Methods", *Rensselaer Polytechnic Institute*, Spring 2009.

[49] S. Geniaut, P. Massin, N. Moës, “A stable 3D contact formulation using X-FEM”, *Revue Européenne de Mécanique Numérique*, pp. 295-275, vol. 16/2 - 2007.

[50] S. Geniaut, P. Massin, N. Moës, D. Colombo, “eXtended Finite Element Method (X-FEM)”, Code_Aster, *EDF-R&D/AMA, ECN, EDF R&D/LaMSID*, 2009.

[51] S. Osher, and R. Fedkiw, "Level Set Methods and Dynamic Implicit Surfaces", *Springer*, 2002.

[52] S. P. Timoshenko, J.N Goodier, “Theory of Elasticity”, 3rd Ed, *McGraw-Hill*, New York, 567 pp., 1970.

[53] Timon Rabczuk, Wolfgang A. Wall, “Extended Finite Element and Meshfree Methods”, *Technical university of Munich*, 2006.

[54] Thomas Menouillard, "Dynamique explicite pour la simulation numérique de propagation de fissure par la méthode des éléments finis étendus (Explicit Dynamics for Numerical Crack Propagation Simulation with X-FEM)", Phd thesis, *LaMCoS, INSA de Lyon*, 2007.

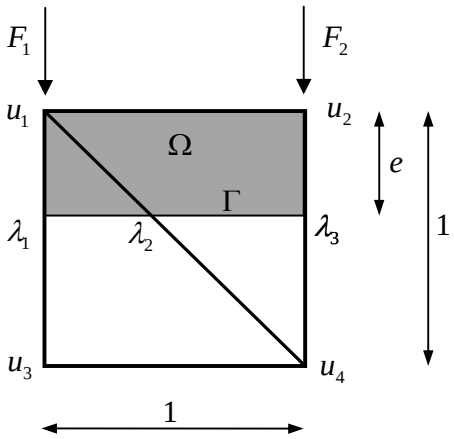
[55] T. Belytschko, C. Parimi, N. Moës, N. Sukumar and S. Usui, “Structured Extended Finite Element Methods for Solids Defined by Implicit Surfaces”, *International Journal for Numerical Methods in Engineering*, Vol. 56, Number 4, pp. 609–635, 2003.

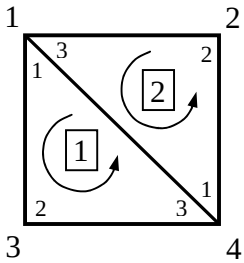
[56] T. Elguedj, A. Gravouil, A. Combescure, “Appropriate extended functions for X-FEM simulation of elastic-plastic fatigue crack growth with frictional contact”, *European Journal of Computational Mechanics*, vol. 15, p. 155-166, 2006.

Appendix A

Calculation for the theory to illustrate the choice of the Lagrange Multiplier space

To prove the theory [29], I am in the middle of all of these computations myself. In fact, it can be easily done by hand.

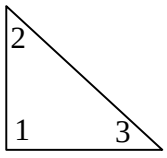




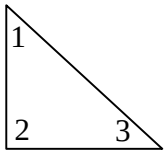
Element	Node 1	Node 2	Node 3
1	1	3	4
2	4	2	1

$$K^e = \begin{array}{c} \begin{matrix} 4 & 2 & 1 \\ 1 & 3 & 4 \end{matrix} \\ \begin{bmatrix} K_{11} & K_{12} & K_{13} \\ K_{21} & K_{22} & K_{23} \\ K_{31} & K_{32} & K_{33} \end{bmatrix} \begin{matrix} 1 & 4 \\ 3 & 2 \\ 4 & 1 \end{matrix} \end{array}$$

The basic FEM given



$$K^e = \frac{1}{2ab}\begin{bmatrix} a^2+b^2 & -b^2 & -a^2 \\ -b^2 & b^2 & 0 \\ -a^2 & 0 & a^2 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 2 & -1 & -1 \\ -1 & 1 & 0 \\ -1 & 0 & 1 \end{bmatrix}$$



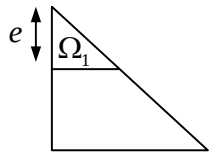
$$K^e = \frac{1}{2ab}\begin{bmatrix} b^2 & -b^2 & 0 \\ -b^2 & a^2+b^2 & -a^2 \\ 0 & -a^2 & a^2 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 1 & -1 & 0 \\ -1 & 2 & -1 \\ 0 & -1 & 1 \end{bmatrix}$$

Assemble the K matrix

$$\begin{bmatrix} K_{11}^1+K_{33}^2 & K_{32}^2 & K_{12}^1 & K_{13}^1+K_{31}^2 \\ K_{23}^2 & K_{22}^2 & 0 & K_{21}^2 \\ K_{21}^1 & 0 & K_{22}^1 & K_{23}^1 \\ K_{31}^1+K_{13}^2 & K_{12}^2 & K_{32}^1 & K_{33}^1+K_{11}^2 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} 2 & -1 & -1 & 0 \\ -1 & 2 & 0 & -1 \\ -1 & 0 & 2 & -1 \\ 0 & -1 & -1 & 2 \end{bmatrix}$$

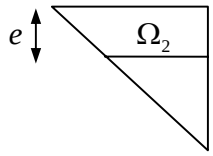
Find the coefficient α, β of the K matrix

- Coefficient α



$$A_{\Omega_1} = \frac{1}{2}e \times e = \frac{e^2}{2}$$
$$A_{\Delta} = \frac{1}{2} \times 1 \times 1 = \frac{1}{2}$$
$$\alpha = \frac{A_{\Omega_1}}{A_{\Delta}} = e^2$$

- Coefficient β



$$A_{\Omega_2} = \frac{1}{2}e \times \left(1+\left(1-e\right)\right) = \frac{e\left(2-e\right)}{2}$$
$$A_{\Delta} = \frac{1}{2}1 \times 1 = \frac{1}{2}$$
$$\beta = \frac{A_{\Omega_2}}{A_{\Delta}} = e\left(2-e\right)$$

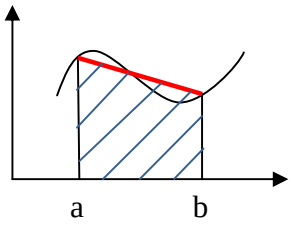
Multiplies the coefficient α, β to the stiffness matrix

$$\begin{bmatrix} \alpha K_{11}^1+\beta K_{33}^2 & \beta K_{32}^2 & \alpha K_{12}^1 & \alpha K_{13}^1+\beta K_{31}^2 \\ \beta K_{23}^2 & \beta K_{22}^2 & 0 & \beta K_{21}^2 \\ \alpha K_{21}^1 & 0 & \alpha K_{22}^1 & \alpha K_{23}^1 \\ \alpha K_{31}^1+\beta K_{13}^2 & \beta K_{12}^2 & \alpha K_{32}^1 & \alpha K_{33}^1+\beta K_{11}^2 \end{bmatrix} =$$

$$= \frac{1}{2}\begin{bmatrix} e^2\times 1+e\left(2-e\right)\times 1 & e\left(2-e\right)\times(-1) & e^2\times(-1) & e^2\times 0+e\left(2-e\right)\times 0 \\ e\left(2-e\right)\times(-1) & e\left(2-e\right)\times 2 & 0 & e\left(2-e\right)\times(-1) \\ e^2\times(-1) & 0 & e^2\times 2 & e^2\times(-1) \\ e^2\times 0+e\left(2-e\right)\times 0 & e\left(2-e\right)\times(-1) & e^2\times(-1) & e^2\times 1+e\left(2-e\right)\times 1 \end{bmatrix}$$

$$= \frac{1}{2}\begin{bmatrix} 2e & e^2-2e & -e^2 & 0 \\ e^2-2e & 4e-2e^2 & 0 & e^2-2e \\ -e^2 & 0 & 2e^2 & -e^2 \\ 0 & e^2-2e & -e^2 & 2e \end{bmatrix}$$

➢ Find the B matrix

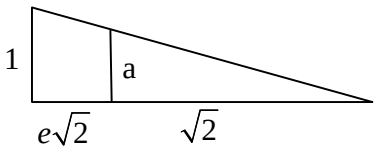


Nodal intergral

❖ With $\varphi_1 = 1,\ \varphi_3 = \varphi_4 = 0$

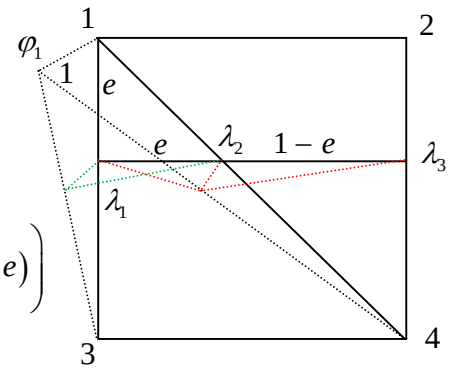
$$\int_\Gamma \lambda_1\varphi_1 = (1-e)\times e\times\frac{1}{2} = \frac{e(1-e)}{2}$$

$$\int_\Gamma \lambda_2\varphi_1 = (1-e)\times e\times\frac{1}{2} + \frac{1}{2}(1-e)^2 = \frac{(1-e)}{2}$$



$$\left(\frac{a}{1} = \frac{\sqrt{2}-e\sqrt{2}}{\sqrt{2}} = (1-e)\right)$$

$$\int_\Gamma \lambda_3\varphi_1 = 0$$

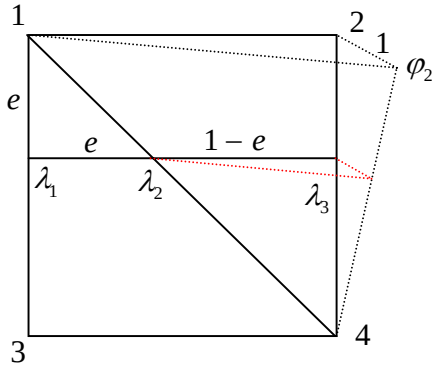


❖ With $\varphi_2 = 1,\ \varphi_1 = \varphi_4 = 0$

$$\int_\Gamma \lambda_1\varphi_2 = 0$$

$$\int_\Gamma \lambda_2\varphi_2 = 0$$

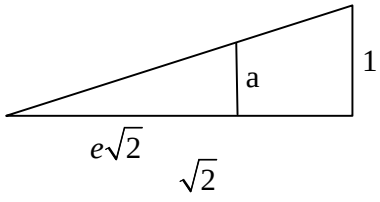
$$\int_\Gamma \lambda_3\varphi_2 = (1-e)\times(1-e)\frac{1}{2} = \frac{(1-e)^2}{2}$$



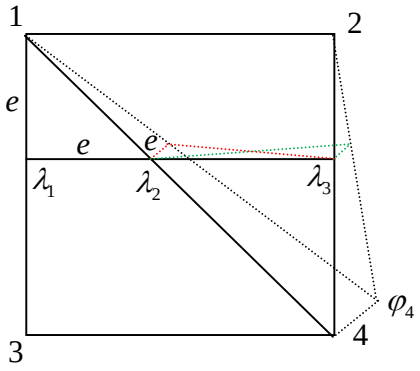
❖ With $\varphi_4 = 1,\ \varphi_1 = \varphi_2 = 0$

$$\int_\Gamma \lambda_1\varphi_4 = 0$$

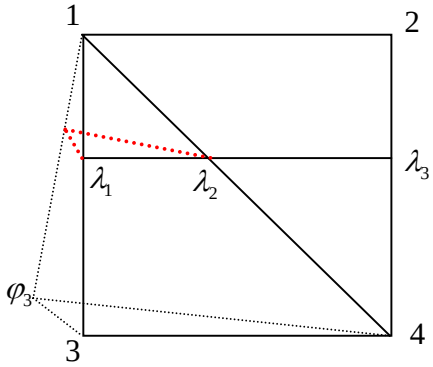
$$\int_\Gamma \lambda_2\varphi_4 = e\times(1-e)\times\frac{1}{2} + e\times e\times\frac{1}{2} = \frac{e}{2}$$



$$\left(\frac{a}{1} = \frac{e\sqrt{2}}{\sqrt{2}} = e\right)$$



$$\int_\Gamma \lambda_3\varphi_4 = e\times(1-e)\frac{1}{2} = \frac{e(1-e)}{2}$$



❖ With $\varphi_3 = 1,\ \varphi_1 = \varphi_4 = 0$

$$\int_\Gamma \lambda_1\varphi_3 = e\times e\times\frac{1}{2} = \frac{e^2}{2}$$

$$\int_\Gamma \lambda_2\varphi_3 = 0$$

$$\int_\Gamma \lambda_3\varphi_3 = 0$$

$$\begin{array}{c}(1)\\(2)\\(3)\\(4)\\ \frac{1}{2}\\(5)\\(6)\\(7)\end{array}\left[\begin{array}{cccc|ccc}
2e & e^2-2e & -e^2 & 0 & -e(1-e) & -(1-e) & 0\\
e^2-2e & 4e-2e^2 & 0 & e^2-2e & 0 & 0 & -(1-e)^2\\
-e^2 & 0 & 2e^2 & -e^2 & -e^2 & 0 & 0\\
0 & e^2-2e & -e^2 & 2e & 0 & -e & -e(1-e)\\
- & - & - & - & - & - & -\\
-e(1-e) & 0 & -e^2 & 0 & 0 & 0 & 0\\
-(1-e) & 0 & 0 & -e & 0 & 0 & 0\\
0 & -(1-e)^2 & 0 & -e(1-e) & 0 & 0 & 0
\end{array}\right]\left[\begin{array}{c}u_1\\u_2\\u_3\\u_4\\-\\\lambda_1\\\lambda_2\\\lambda_3\end{array}\right] = \left[\begin{array}{c}-F_1\\-F_2\\0\\0\\-\\0\\0\\0\end{array}\right] \qquad \text{(A.1)}$$

(7) $-(1-e)^2 u_2 - e(1-e)u_4 = 0$

$$u_4 = -\left(\frac{1-e}{e}\right)u_2 \quad \Rightarrow u_2 = -\left(\frac{e}{1-e}\right)u_4 \qquad \text{(A.2)}$$

(6) $-(1-e)u_1 - eu_4 = 0$

$$u_1 = -\left(\frac{e}{(1-e)}\right)u_4 \qquad \text{(A.3)}$$

(5) $-e(1-e)u_1 - e^2u_3 = 0$

$$u_3 = -\frac{(1-e)}{e}u_1 = +u_4 \qquad \text{(A.4)}$$

(2) $(e^2-2e)u_1 + (4e-2e^2)u_2 + (e^2-2e)u_4 - (1-e)^2\lambda_3 = -2F_2$

$$-\frac{e(e^2-2e)}{(1-e)}u_4 - \frac{e(4e-2e^2)}{1-e}u_4 + (e^2-2e)u_4 - (1-e)^2\lambda_3 = -2F_2$$

$$\left(\frac{-e^3+2e^2-4e^2+2e^3+e^2-2e-e^3+2e^2}{1-e}\right)u_4 - (1-e)^2\lambda_3 = -2F_2$$

$$\left(\frac{e^2-2e}{1-e}\right)u_4-(1-e)^2\lambda_3=-2F_2$$

$$\lambda_3=\frac{1}{(1-e)^2}\left(\frac{e^2-2e}{1-e}\right)u_4+2F_2 \tag{A.5}$$

(3) $-e^2u_1+2e^2u_3-e^2u_4-e^2\lambda_1=0$

$$\frac{e^3}{(1-e)}u_4+2e^2u_4-e^2u_4-e^2\lambda_1=0$$

$$\frac{e^3+e^2-e^3}{1-e}u_4-e^2\lambda_1=0$$

$$\lambda_1=\frac{1}{(1-e)}u_4 \tag{A.6}$$

(4) $(e^2-2e)u_2-e^2u_3+2eu_4-e\lambda_2-e(1-e)\lambda_3=0$

$$-\frac{e(e^2-2e)}{(1-e)}u_4+(2e-e^2)u_4-e\lambda_2-e(1-e)\frac{1}{(1-e)^2}\left[\frac{e^2-2e}{1-e}u_4+2F_2\right]=0$$

$$\frac{(-e^3+2e^2)(1-e)+(2e-e^2)(1-2e+e^2)-e^3+2e^2}{(1-e)^2}u_4-e\lambda_2-\frac{2e}{1-e}F_2=0$$

$$\frac{2e-e^2}{(1-e)^2}u_4-e\lambda_2-\frac{2e}{1-e}F_2=0$$

$$\lambda_2=\frac{2-e}{(1-e)^2}u_4-\frac{2}{(1-e)}F_2 \tag{A.7}$$

(1) $2eu_1+(e^2-2e)u_2-e^2u_3-e(1-e)\lambda_1-(1-e)\lambda_2=-2F_1$

$$-\frac{2e^2}{(1-e)}u_4-\frac{e(e^2-2e)}{(1-e)}u_4-e^2u_4-e(1-e)\frac{1}{(1-e)}u_4-(1-e)\lambda_2=-2F_1$$

$$\frac{-2e^2-e^3+2e^2-e^2+e^3-e+e^2}{(1-e)}u_4-(1-e)\lambda_2=-2F_1$$

$$\frac{-e}{(1-e)}u_4-(1-e)\left[\frac{2-e}{(1-e)^2}u_4-\frac{2}{1-e}F_2\right]=-2F_1$$

$$\frac{-2}{(1-e)}u_4+2F_2=-2F_1$$

$$u_4=\frac{(1-e)}{2}(2F_1+2F_2)$$

$$u_4=(1-e)(F_1+F_2) \tag{A.8}$$

Replaced to the formula of $\lambda_1, \lambda_2, \lambda_3$ in the previous calculation:

$$\lambda_1 = \frac{1}{(1-e)} u_4 = \frac{1}{(1-e)}(1-e)(F_1+F_2) = (F_1+F_2)$$

$$\lambda_2 = \frac{2-e}{(1-e)^2} u_4 - \frac{2}{(1-e)} F_2 = \frac{2-e}{(1-e)^2}(1-e)(F_1+F_2) - \frac{2}{(1-e)} F_2$$

$$\lambda_2 = \frac{(1-e)}{(1-e)}(F_1+F_2) + \frac{1}{(1-e)}(F_1+F_2) - \frac{2}{(1-e)} F_2 = (F_1+F_2) + \frac{1}{(1-e)}(F_1-F_2)$$

$$\lambda_3 = \frac{1}{(1-e)^2}\left[\frac{e^2-2e}{1-e} u_4 + 2F_2\right]$$

$$\lambda_3 = \frac{1}{(1-e)^2}\left[(e^2-2e)(F_1+F_2) + 2F_2\right]$$

$$\lambda_3 = \frac{1}{(1-e)^2}\left[(1-2e+e^2)(F_1+F_2) - F_1 - F_2 + 2F_2\right]$$

$$\lambda_3 = (F_1+F_2) - \frac{1}{(1-e)^2}(F_1-F_2)$$

In summary, we have

$$\begin{aligned} F &= F_1 + F_2 \\ [F] &= F_1 - F_2 \\ \lambda_1 &= F \\ \lambda_2 &= F + \frac{1}{(1-e)}[F] \\ \lambda_3 &= F - \frac{1}{(1-e)^2}[F] \end{aligned}$$

- Set $\lambda_2 = \lambda_3$, the system of equations in (A.1) given

$$\begin{array}{c} (1) \\ (2) \\ (3) \\ (4) \\ - \\ (5) \\ (6) \end{array} \left[\begin{array}{ccccc|cc} 2e & e^2-2e & -e^2 & 0 & | & -e(1-e) & -(1-e) \\ e^2-2e & 4e-2e^2 & 0 & e^2-2e & | & 0 & -(1-e)^2 \\ -e^2 & 0 & 2e^2 & -e^2 & | & -e^2 & 0 \\ 0 & e^2-2e & -e^2 & 2e & | & 0 & e^2-2e \\ - & - & - & - & - & - & - \\ -e(1-e) & 0 & -e^2 & 0 & | & 0 & 0 \\ -(1-e) & -(1-e)^2 & 0 & e^2-2e & | & 0 & 0 \end{array}\right] \times \left[\begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ - \\ \lambda_1 \\ \lambda_2 \end{array}\right] = \left[\begin{array}{c} -2F_1 \\ -2F_2 \\ 0 \\ 0 \\ - \\ 0 \\ 0 \end{array}\right] \qquad \text{(A.9)}$$

(5) $-e(1-e)u_1 - e^2 u_3 = 0$

$$u_3 = -\frac{(1-e)}{e}u_1 \tag{A.10}$$

(6) $-(1-e)u_1 - (1-e)^2 u_2 + (e^2-2e)u_4 = 0$

$$u_4 = \frac{(1-e)}{(e^2-2e)}u_1 + \frac{(1-e)^2}{(e^2-2e)}u_2 \tag{A.11}$$

(4) $(e^2-2e)u_2 - e^2 u_3 + 2eu_4 + (e^2-2e)\lambda_2 = 0$

$$(e^2-2e)u_2 + e(1-e)u_1 + \frac{2e(1-e)}{(e^2-2e)}u_1 + \frac{2e(1-e)^2}{(e^2-2e)}u_2 + (e^2-2e)\lambda_2 = 0$$

$$\frac{(e-e^2)(e^2-2e)+2e-2e^2}{(e^2-2e)}u_1 + \frac{e^4-4e^3+4e^2+2e-4e^2+2e^3}{(e^2-2e)}u_2 + (e^2-2e)\lambda_2 = 0$$

$$\frac{-e^4+3e^3-4e^2+2e}{(e^2-2e)}u_1 + \frac{e^4-2e^3+2e}{(e^2-2e)}u_2 + (e^2-2e)\lambda_2 = 0$$

$$\lambda_2 = -\frac{-e^4+3e^3-4e^2+2e}{(e^2-2e)^2}u_1 - \frac{e^4-2e^3+2e}{(e^2-2e)^2}u_2 \tag{A.12}$$

(3) $-e^2 u_1 + 2e^2 u_3 - e^2 u_4 - e^2\lambda_1 = 0$

$$-e^2 u_1 + 2e(1-e)u_1 - \frac{e^2(1-e)}{(e^2-2e)}u_1 - \frac{e^2(1-e)^2}{(e^2-2e)}u_2 - e^2\lambda_1 = 0$$

$$\frac{-e^4+2e^3-2e^3+4e^2+2e^4-4e^3-e^2+e^3}{(e^2-2e)}u_1 - \frac{e^2(1-e)^2}{(e^2-2e)}u_2 - e^2\lambda_1 = 0$$

$$\lambda_1 = \frac{e^2-3e+3}{(e^2-2e)}u_1 - \frac{(1-e)^2}{(e^2-2e)}u_2 \tag{A.13}$$

(2) $(e^2-2e)u_1 + (4e-2e^2)u_2 + (e^2-2e)u_4 - (1-e)^2\lambda_2 = -2F_2$

$$(e^2-2e)u_1 + (4e-2e^2)u_2 + (1-e)u_1 + (1-e)^2 u_2 - (1-e)^2\lambda_2 = -2F_2$$

$$(e^2-3e+1)u_1 + (-e^2+2e+1)u_2 - (1-e)^2\lambda_2 = -2F_2$$

$$(e^2-3e+1)u_1 + (-e^2+2e+1)u_2 + (1-e)^2\frac{(-e^4+3e^3-4e^2+2e)}{(e^2-2e)^2}u_1 + (1-e)^2\frac{(e^4-2e^3+2e)}{(e^2-2e)^2}u_2 = -2F_2$$

$$\frac{(-2e^5+6e^4-3e^3-4e^2+2e)}{(e^2-2e)^2}u_1 + \frac{(2e^5-6e^4+4e^3+2e)}{(e^2-2e)^2}u_2 = -2F_2$$

$$u_1 = -\frac{\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}u_2 - \frac{2\left(e^2 - 2e\right)^2}{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}F_2$$

$$u_1 = -\frac{\left(2e^4 - 6e^3 + 4e^2 + 2\right)}{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}u_2 - \frac{2e\left(e-2\right)^2}{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}F_2 \qquad \text{(A.14)}$$

(1) $2eu_1 + \left(e^2 - 2e\right)u_2 - e^2u_3 - e\left(1-e\right)\lambda_1 - \left(1-e\right)\lambda_2 = -2F_1$

$$2eu_1 + \left(e^2 - 2e\right)u_2 + e\left(1-e\right)u_1 - e\left(1-e\right)\frac{\left(e^2 - 3e + 3\right)}{\left(e^2 - 2e\right)}u_1 + \frac{e\left(1-e\right)^3}{\left(e^2 - 2e\right)}u_2$$

$$+\left(1-e\right)\frac{\left(-e^4 + 3e^3 - 4e^2 + 2e\right)}{\left(e^2 - 2e\right)^2}u_1 + \left(1-e\right)\frac{\left(e^4 - 2e^3 + 2e\right)}{\left(e^2 - 2e\right)^2}u_2 = -2F_1$$

$$\frac{\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(e^2 - 2e\right)^2}u_1 + \frac{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}{\left(e^2 - 2e\right)^2}u_2 = -2F_1$$

$$-\frac{\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(e^2 - 2e\right)^2} \times \frac{\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}u_2 -$$

$$-\frac{2\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}F_2 + \frac{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}{\left(e^2 - 2e\right)^2}u_2 = -2F_1$$

$$\frac{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)^2 - \left(2e^5 - 6e^4 + 4e^3 + 2e\right)^2}{\left(e^2 - 2e\right)^2 \times \left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}u_2 = -2F_1 + \frac{2\left(2e^5 - 6e^4 + 4e^3 + 2e\right)}{\left(-2e^5 + 6e^4 - 3e^3 - 4e^2 + 2e\right)}F_2$$

$$u_2 = \frac{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}{\left(4e^3 - 12e^2 + 7e + 4\right)}2F_1 - \frac{\left(2e^4 - 6e^3 + 4e^2 + 2\right)}{\left(4e^3 - 12e^2 + 7e + 4\right)}2F_2 \qquad \text{(A.15)}$$

Thus,

$$u_1 = -\frac{\left(2e^4 - 6e^3 + 4e^2 + 2\right)}{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}\left[\frac{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}{\left(4e^3 - 12e^2 + 7e + 4\right)}2F_1 - \frac{\left(2e^4 - 6e^3 + 4e^2 + 2\right)}{\left(4e^3 - 12e^2 + 7e + 4\right)}2F_2\right] -$$

$$-\frac{2e\left(e-2\right)^2}{\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}F_2 \qquad \text{(A.16)}$$

Consequently,

$$\lambda_1 = \frac{e^2 - 3e + 3}{\left(e^2 - 2e\right)}u_1 - \frac{\left(1-e\right)^2}{\left(e^2 - 2e\right)}u_2$$

$$\lambda_1 = -\frac{\left(e^2 - 3e + 3\right)\left(2e^4 - 6e^3 + 4e^2 + 2\right) + \left(1-e\right)^2\left(-2e^4 + 6e^3 - 3e^2 - 4e + 2\right)}{\left(e^2 - 2e\right)\left(4e^3 - 12e^2 + 7e + 4\right)}2F_1 +$$

$$+\frac{\left(e^2-3e+3\right)\left(2e^4-6e^3+4e^2+2\right)^2-\left(e^2-3e+3\right)(e-2)\left(e^2-2e\right)\left(4e^3-12e^2+7e+4\right)}{\left(e^2-2e\right)\left(4e^3-12e^2+7e+4\right)\left(-2e^4+6e^3-3e^2-4e+2\right)}2F_2$$

$$+\frac{(1-e)^2\left(2e^4-6e^3+4e^2+2\right)\left(-2e^4+6e^3-3e^2-4e+2\right)}{\left(e^2-2e\right)\left(4e^3-12e^2+7e+4\right)\left(-2e^4+6e^3-3e^2-4e+2\right)}2F_2$$

$$\lambda_1 = I_1 + I_2$$

$$I_1 = -\frac{\left(e^2-3e+3\right)\left(2e^4-6e^3+4e^2+2\right)+(1-e)^2\left\{-\left[\left(2e^4-6e^3+4e^2+2\right)-(e-2)^2\right]\right\}}{e(e-2)\left(4e^3-12e^2+7e+4\right)}2F_1$$

$$I_1 = F_1 - \frac{\left(2e^3-9e^2+14e-8\right)}{e\left(4e^3-12e^2+7e+4\right)}F_1$$

$$I_2 = \frac{(e-2)\left(-2e^4+6e^3-3e^2-4e+2\right)\left(2e^4-5e^3-e^2+9e-4\right)}{e(e-2)\left(4e^3-12e^2+7e+4\right)\left(-2e^4+6e^3-3e^2-4e+2\right)}2F_2$$

$$I_2 = F_2 + \frac{\left(2e^3-9e^2+14e-8\right)}{e\left(4e^3-12e^2+7e+4\right)}F_2$$

$$\lambda_1 = F_1 - \frac{\left(2e^3-9e^2+14e-8\right)}{e\left(4e^3-12e^2+7e+4\right)}F_1 + F_2 + \frac{\left(2e^3-9e^2+14e-8\right)}{e\left(4e^3-12e^2+7e+4\right)}F_2$$

$$\lambda_1 = \left(F_1+F_2\right) - \frac{\left(2e^3-9e^2+14e-8\right)}{\left(4e^3-12e^2+7e+4\right)}\frac{1}{e}\left(F_1-F_2\right) \qquad \text{(A.17)}$$

Replaced u_1, u_2 to λ_2

$$\lambda_2 = -\frac{-e^4+3e^3-4e^2+2e}{\left(e^2-2e\right)^2}u_1 - \frac{e^4-2e^3+2e}{\left(e^2-2e\right)^2}u_2$$

$$\lambda_2 = -\frac{-e^3+3e^2-4e+2}{e\left(e-2\right)^2}u_1 - \frac{e^3-2e^2+2}{e\left(e-2\right)^2}u_2 \qquad \text{(A.18)}$$

The coefficient of F_1

$$= \frac{\left(2e^6-15e^5+42e^4-52e^3+24e^2\right)}{e\left(e-2\right)^2\left(4e^3-12e^2+7e+4\right)}2F_1$$

$$= \frac{4e^5-30e^4+84e^3-104e^2+48e}{\left(e-2\right)^2\left(4e^3-12e^2+7e+4\right)}F_1$$

$$= \frac{(e-2)^2\left[\left(4e^3-12e^2+7e+4\right)-\left(2e^2-5e+4\right)\right]}{(e-2)^2\left(4e^3-12e^2+7e+4\right)}F_1$$

$$= F_1 - \frac{\left(2e^2-5e+4\right)}{\left(4e^3-12e^2+7e+4\right)}F_1$$

The coefficient of F_2

$$= \frac{\left(4e^3-10e^2+2e+8\right)(e-2)^2\,e\left(-2e^4+6e^3-3e^2-4e+2\right)}{e(e-2)^2\left(-2e^4+6e^3-3e^2-4e+2\right)\left(4e^3-12e^2+7e+4\right)}F_2$$

$$= \frac{\left[\left(4e^3-12e^2+7e+4\right)+\left(2e^2-5e+4\right)\right]}{\left(4e^3-12e^2+7e+4\right)}F_2$$

$$= F_2 + \frac{\left(2e^2-5e+4\right)}{\left(4e^3-12e^2+7e+4\right)}F_2$$

Then

$$\lambda_2 = (F_1+F_2) - \frac{\left(2e^2-5e+4\right)}{\left(4e^3-12e^2+7e+4\right)}(F_1-F_2) \tag{A.19}$$

As a consequence,

$$\lambda_1 = F - \frac{\left(2e^3-9e^2+14e-8\right)}{\left(4e^3-12e^2+7e+4\right)}\frac{1}{e}[F]$$

$$\lambda_2 = \lambda_3 \tag{A.20}$$

$$\lambda_3 = F - \frac{\left(2e^2-5e+4\right)}{\left(4e^3-12e^2+7e+4\right)}[F]$$

- We set $\lambda_2 = (1-e)\lambda_1 + e\lambda_3$, the system of equations (A.1) returns:

$$\begin{array}{c}(1)\\(2)\\(3)\\(4)\\-\\(5)\\(6)\end{array}\left[\begin{array}{ccccc|cc} 2e & e^2-2e & -e^2 & 0 & | & (e-1) & -e(1-e) \\ e^2-2e & 4e-2e^2 & 0 & e^2-2e & | & 0 & -(1-e)^2 \\ -e^2 & 0 & 2e^2 & -e^2 & | & -e^2 & 0 \\ 0 & e^2-2e & -e^2 & 2e & | & -e(1-e) & -e \\ - & - & - & - & - & - & - \\ (e-1) & 0 & -e^2 & -e(1-e) & | & 0 & 0 \\ -e(1-e) & -(1-e)^2 & 0 & -e & | & 0 & 0 \end{array}\right] \times \left[\begin{array}{c} u_1 \\ u_2 \\ u_3 \\ u_4 \\ - \\ \lambda_1 \\ \lambda_3 \end{array}\right] = \left[\begin{array}{c} -2F_1 \\ -2F_2 \\ 0 \\ 0 \\ - \\ 0 \\ 0 \end{array}\right] \tag{A.21}$$

Take the first line of the B matrix

$$
\begin{aligned}
&-e(1-e)\lambda_1-(1-e)\lambda_2+0\lambda_3\\
&=-e(1-e)\lambda_1-(1-e)\left[(1-e)\lambda_1+e\lambda_3\right]\\
&=\left[-e(1-e)-(1-e)^2\right]\lambda_1-e(1-e)\lambda_3\\
&\left.\begin{aligned}\lambda_1^*&=\lambda_1+(1-e)\lambda_2\\ \lambda_3^*&=\lambda_3+e\lambda_2\end{aligned}\right\rangle -e.e-e(1-e)=-e
\end{aligned}
$$

$(5)\quad (e-1)u_1-e^2u_3-e(1-e)u_4=0$

$$u_1=\frac{e^2}{(e-1)}u_3-eu_4 \tag{A.22}$$

$(6)\quad -e(1-e)u_1-(1-e)^2u_2-eu_4=0$

$$
\begin{aligned}
&-e(1-e)\left[\frac{e^2}{(e-1)}u_3-eu_4\right]-(1-e)^2u_2-eu_4=0\\
&e^3u_3+e^2(1-e)u_4-(1-e)^2u_2-eu_4=0\\
&e^3u_3+e\left(-e^2+e-1\right)u_4-(1-e)^2u_2=0\\
&u_2=\frac{e^3}{(1-e)^2}u_3+\frac{e\left(-e^2+e-1\right)}{(1-e)^2}u_4
\end{aligned}
\tag{A.23}
$$

$(3)\quad -e^2u_1+2e^2u_3-e^2u_4-e^2\lambda_1=0$

$$
\begin{aligned}
&-\frac{e^2}{(e-1)}u_3+eu_4+2u_3-u_4-\lambda_1=0\\
&\frac{-e^2+2e-2}{(e-1)}u_3+(e-1)u_4-\lambda_1=0\\
&u_3=-\frac{(e-1)^2}{\left(-e^2+2e-2\right)}u_4+\frac{(e-1)}{\left(-e^2+2e-2\right)}\lambda_1
\end{aligned}
\tag{A.24}
$$

$(2)\quad \left(e^2-2e\right)u_1+\left(4e-2e^2\right)u_2+\left(e^2-2e\right)u_4-(1-e)^2\lambda_3=-2F_2$

$$\left(e^2-2e\right)\left[\frac{e^2}{(e-1)}u_3-eu_4\right]+\left(4e-2e^2\right)\left[\frac{e^3}{(1-e)^2}u_3+\frac{e\left(-e^2+e-1\right)}{(1-e)^2}u_4\right]+\left(e^2-2e\right)u_4-(1-e)^2\lambda_3=-2F_2$$

$$\frac{\left(-e^5+e^4+2e^3\right)}{(1-e)^2}u_3+\frac{\left(e^5-e^4-3e^3+3e^2-2e\right)}{(1-e)^2}u_4-(1-e)^2\lambda_3=-2F_2$$

$$\lambda_3 = \frac{2F_2}{(1-e)^2} + \frac{\left(-e^5+e^4+2e^3\right)}{(1-e)^4}u_3 + \frac{\left(e^5-e^4-3e^3+3e^2-2e\right)}{(1-e)^4}u_4$$

$$\lambda_3 = \frac{2F_2}{(1-e)^2} - \frac{\left(-e^5+e^4+2e^3\right)}{(1-e)^4}\frac{(e-1)^2}{\left(-e^2+2e-2\right)}u_4 + \frac{\left(-e^5+e^4+2e^3\right)}{(1-e)^4}\frac{(e-1)}{\left(-e^2+2e-2\right)}\lambda_1$$
$$+\frac{\left(e^5-e^4-3e^3+3e^2-2e\right)}{(1-e)^4}u_4$$

$$\lambda_3 = \frac{2F_2}{(1-e)^2} + \frac{\left(e^5-e^4-2e^3\right)}{(1-e)^3\left(-e^2+2e-2\right)}\lambda_1 + \frac{\left(-4e^4+12e^3-10e^2+4e\right)}{(1-e)^4\left(-e^2+2e-2\right)}u_4 \qquad \text{(A.25)}$$

$(4)\quad \left(e^2-2e\right)u_2 - e^2u_3 + 2eu_4 - e(1-e)\lambda_1 - e\lambda_3 = 0$

Replaced u_2, u_3, λ_3 into (4)

$$\left(e^2-2e\right)\left[\frac{e^3}{(1-e)^2}u_3 + \frac{e\left(-e^2+e-1\right)}{(1-e)^2}u_4\right] - e^2u_3 + 2eu_4 - e(1-e)\lambda_1 -$$
$$-\left[\frac{2F_2}{(1-e)^2} + \frac{\left(e^5-e^4-2e^3\right)}{(1-e)^3\left(-e^2+2e-2\right)}\lambda_1 + \frac{\left(-4e^4+12e^3-10e^2+4e\right)}{(1-e)^4\left(-e^2+2e-2\right)}u_4\right]e = 0$$

$$\left(e^2-2e\right)\left\{\frac{e^3}{(1-e)^2}\left[-\frac{(e-1)^2}{\left(-e^2+2e-2\right)}u_4 + \frac{(e-1)}{\left(-e^2+2e-2\right)}\lambda_1\right] + \frac{e\left(-e^2+e-1\right)}{(1-e)^2}u_4\right\} -$$
$$-e^2\left[-\frac{(e-1)^2}{\left(-e^2+2e-2\right)}u_4 + \frac{(e-1)}{\left(-e^2+2e-2\right)}\lambda_1\right] + 2eu_4 - e(1-e)\lambda_1 -$$
$$-\left[\frac{2F_2}{(1-e)^2} + \frac{\left(e^5-e^4-2e^3\right)}{(1-e)^3\left(-e^2+2e-2\right)}\lambda_1 + \frac{\left(-4e^4+12e^3-10e^2+4e\right)}{(1-e)^4\left(-e^2+2e-2\right)}u_4\right]e = 0$$

$$-\frac{2eF_2}{(1-e)^2} + \frac{\left(-2e^6+8e^5-14e^4+17e^3-9e^2+2e\right)}{(1-e)^3\left(-e^2+2e-2\right)}\lambda_1 + \frac{\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}{(1-e)^4\left(-e^2+2e-2\right)}u_4 = 0$$

$$u_4 = -\frac{(1-e)\left(-2e^6+8e^5-14e^4+17e^3-9e^2+2e\right)}{\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}\lambda_1 + \frac{2e(1-e)^2\left(-e^2+2e-2\right)F_2}{\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)} \qquad \text{(A.26)}$$

Replaced u_4 to λ_3

$$\lambda_3 = \frac{2F_2}{(1-e)^2} + \frac{\left(e^5-e^4-2e^3\right)}{(1-e)^3\left(-e^2+2e-2\right)}\lambda_1 +$$

$$+\frac{\left(-4e^4+12e^3-10e^2+4e\right)}{(1-e)^4\left(-e^2+2e-2\right)}\left[-\frac{(1-e)\left(-2e^6+8e^5-14e^4+17e^3-9e^2+2e\right)}{\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}\lambda_1+\right.$$
$$\left.+\frac{2e(1-e)^2\left(-e^2+2e-2\right)F_2}{\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}\right]$$

$$\lambda_3=\frac{4e^6-24e^5+54e^4-60e^3+34e^2-8e}{(1-e)^2\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}F_2+$$
$$+\frac{\left(2e^{11}-18e^{10}+75e^9-191e^8+327e^7-389e^6+320e^5-174e^4+56e^3-8e^2\right)}{(1-e)^3\left(-e^2+2e-2\right)\left(2e^6-8e^5+15e^4-20e^3+13e^2-4e\right)}\lambda_1 \qquad \text{(A.27)}$$

$$(1)\quad 2eu_1+\left(e^2-2e\right)u_2-e^2u_3+(e-1)\lambda_1-e(1-e)\lambda_3=-2F_1$$

Replaced by u_1, u_2 and u_3

$$\frac{\left(e^5-2e^3\right)}{(1-e)^2}u_3+\frac{\left(e^7-2e^6-e^5+6e^4-6e^3+e^2\right)}{(1-e)^2\left(-e^2+2e-2\right)}u_4+\frac{\left(-2e^3+4e^2-4e+2\right)}{\left(-e^2+2e-2\right)}\lambda_1-e(1-e)\lambda_3=-2F_1$$

Replaced by u_3 and λ_3

$$\frac{\left(2e^5-6e^4+5e^3-3e^2\right)}{(1-e)^3\left(-e^2+2e-2\right)}u_4+\frac{\left(-2e^5+8e^4-12e^3+14e^2-8e+2\right)}{(1-e)^2\left(-e^2+2e-2\right)}\lambda_1-\frac{2eF_2}{(1-e)}=-2F_1$$

Replaced by u_4

$$\lambda_1=-\frac{\left(-e^4+3e^3-4e^2+2e\right)(1-e)\left(-4e^5+16e^4-26e^3+24e^2-8e\right)}{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)\left(-e^4+3e^3-4e^2+2e\right)(1-e)}F_2-$$
$$-\frac{\left(-e^4+3e^3-4e^2+2e\right)(1-e)\left(-4e^5+16e^4-30e^3+40e^2-26e+8\right)}{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)\left(-e^4+3e^3-4e^2+2e\right)(1-e)}F_1$$

$$\lambda_1=\frac{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)-\left(2e^3-8e^2+9e-4\right)}{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)}F_2+$$
$$+\frac{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)+\left(2e^3-8e^2+9e-4\right)}{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)}F_1$$

$$\lambda_1=F_1+F_2+\frac{\left(2e^3-8e^2+9e-4\right)}{\left(4e^5-16e^4+28e^3-32e^2+17e-4\right)}\left(F_1-F_2\right) \qquad \text{(A.28)}$$

Calculate λ_3

From the equation (A.27)

$$\lambda_3 = \frac{\left(4e^6 - 24e^5 + 54e^4 - 60e^3 + 34e^2 - 8e\right)}{(1-e)^2\left(2e^6 - 8e^5 + 15e^4 - 20e^3 + 13e^2 - 4e\right)} F_2 +$$

$$+ \frac{\left(2e^{11} - 18e^{10} + 75e^9 - 191e^8 + 327e^7 - 389e^6 + 320e^5 - 174e^4 + 56e^3 - 8e^2\right)}{(1-e)^3\left(-e^2 + 2e - 2\right)\left(2e^6 - 8e^5 + 15e^4 - 20e^3 + 13e^2 - 4e\right)} \lambda_1$$

$$\lambda_3 = \frac{(1-e)^2\left(4e^3 - 16e^2 + 18e - 8\right)e}{(1-e)^2\left(2e^5 - 8e^4 + 15e^3 - 20e^2 + 13e - 4\right)e} F_2 +$$

$$+ \frac{(1-e)^3\left(-e^2 + 2e - 2\right)\left(2e^5 - 8e^4 + 13e^3 - 12e^2 + 4e\right)e}{(1-e)^3\left(-e^2 + 2e - 2\right)\left(2e^5 - 8e^4 + 15e^3 - 20e^2 + 13e - 4\right)e} \lambda_1$$

$$\lambda_3 = \frac{\left(4e^3 - 16e^2 + 18e - 8\right)}{\left(2e^5 - 8e^4 + 15e^3 - 20e^2 + 13e - 4\right)} F_2 +$$

$$+ \frac{\left(2e^5 - 8e^4 + 13e^3 - 12e^2 + 4e\right)}{\left(2e^5 - 8e^4 + 15e^3 - 20e^2 + 13e - 4\right)}\left[F_1 + F_2 + \frac{\left(2e^3 - 8e^2 + 9e - 4\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}\left(F_1 - F_2\right)\right]$$

$$\lambda_3 = \frac{\left(4e^5 - 16e^4 + 26e^3 - 24e^2 + 8e\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)} F_1 + \frac{\left(4e^5 - 16e^4 + 30e^3 - 40e^2 + 26e - 8\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)} F_2$$

$$\lambda_3 = \frac{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}\left(F_1 + F_2\right) - \frac{\left(2e^3 - 8e^2 + 9e - 4\right)F_1 - \left(2e^3 - 8e^2 + 9e - 4\right)F_2}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}$$

$$\lambda_3 = \left(F_1 + F_2\right) - \frac{\left(2e^3 - 8e^2 + 9e - 4\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}\left(F_1 - F_2\right) \qquad \text{(A.29)}$$

With $\lambda_2 = (1-e)\lambda_1 + e\lambda_3$

$$\lambda_1 = F + \frac{\left(2e^3 - 8e^2 + 9e - 4\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}\left[F\right]$$

$$\lambda_2 = (1-e)\lambda_1 + e\lambda_3 \qquad \text{(A.30)}$$

$$\lambda_3 = F - \frac{\left(2e^3 - 8e^2 + 9e - 4\right)}{\left(4e^5 - 16e^4 + 28e^3 - 32e^2 + 17e - 4\right)}\left[F\right]$$

Appendix B

Algorithm to design the Lagrange Multiplier space

This appendix will provide additional details concerning the use of three algorithms.

B.1 Description of the Moës algorithm (algo1)

The algorithm introduced by Moës [29] is presented in the problem of how to impose Dirichlet conditions on an interface in the context of X-FEM. It shows that the technique of Lagrange multipliers to impose Dirichlet conditions must be used carefully, because the inf-sup condition is not always respected. The paper is restricted to the 2D case, but the algorithm presented is easily generalizable to 3D case. The first phase is a phase of selection of nodes, in which selected nodes are those "important" for the approximation of the Lagrange multipliers. The other nodes are excessive and lead to the oscillations of the Lagrange multipliers. Once the nodes "important" selected, relationships of equality and linear combinations are placed between the Lagrange multipliers, to narrow the space of multipliers. Thus, the Lagrange multipliers of the edge from a same selected node are equal, and if a Lagrange multiplier is on an edge whose two end nodes are selected, then it is linked to the two Lagrange connected closest.

More formally, *E* and *N* are the sets containing all the edges and all nodes of the mesh. The two ends of an edge $e \in E$ are noted $\left(v_1(e), v_2(e)\right) \in N^2$. First, it begins with an initialization step (iteration $k = 0$ of the algorithm). It determines S_e^0, the set of edges that are cut by the interface. The interface is represented by the normal level set *lsn*, an edge $e \in E$ is cut by the interface if and only if $lsn\left(v_1(e)\right) \cdot lsn\left(v_2(e)\right) \le 0$. Note that if the interface coincides with the node $v_1(e)$ or node $v_2(e)$, the edge e is belong to S_e^0:

$$S_e^0 = \left\{ \mathrm{e} \in E, \quad lsn\left(v_1(e)\right) \cdot lsn\left(v_2(e)\right) \le 0 \right\} \tag{B.1}$$

Consider N_e the set of nodes connected by the elements S_e^0:

$$N_e = \left\{ n \in N, \quad \exists e \in S_e^0 \quad n = v_1(e) \quad \text{or} \quad n = v_2(e) \right\} \tag{B.2}$$

Consider S_n^0 the set of nodes selected in the iteration $k = 0$ (initialization). These nodes are those that coincide with the interface (this set can be empty):

$$S_n^0 = \left\{ n \in N_e, \quad lsn(n) = 0 \right\} \tag{B.3}$$

After this initialization phase, the algorithm iterates of $k = 1, \mathrm{nmax\_iter}$.

At each iteration, we perform the following steps:

- Updated all set of edges: it removes those that are connected to a node selected in previous iteration

$$S_e^k = S_e^{k-1} \setminus \left\{ \mathrm{e} \in S_e^{k-1}, \quad v_1(e) \in S_n^{k-1} \text{ or } v_2(e) \in S_n^{k-1} \right\} \tag{B.4}$$

- Calculated the score of these nodes: for each node in N_e, we calculate a score composed of 2 numbers: the first is the number of edges in S_e^k connected, and the second is the absolute value of the normal level set in that node. This score sc_no is a matrix with two columns that are lines represent the node.

$$\forall n \in N_e \quad \begin{cases} sc\_no^k(n,1) = number\ of\ edge\ connect\ at\ node\ n \\ sc\_no^k(n,2) = \left| lsn(n) \right| \end{cases} \tag{B.5}$$

- Calculated the score of these edges: for each edge in S_e^k, we calculate a score composed of 2 numbers: the first digit corresponds to the absolute value of the difference of the 1st digit of the score of 2 nodes end and the second is a relationship between the values of the 2nd digit of 2 nodes ends (ie. a ratio of values lsn. This score sc_ar is a matrix with two columns that are the lines represent the edge.

$$\forall e \in S_e^k, \quad \forall j \in \{1,2\}, \quad s_j = sc\_no^k\left(v_j(e),1\right), \quad l_j = sc\_no^k\left(v_j(e),2\right)$$

$$\begin{cases} sc\_ar^k(e,1) = \left| s_1 - s_2 \right| \\ sc\_ar^k(e,2) = \begin{cases} \dfrac{l_1}{l_1 + l_2} & \text{if} \quad s_1 > s_2 \\ \dfrac{l_1}{l_1 + l_2} & \text{if} \quad s_1 < s_2 \\ \dfrac{\min(l_1, l_2)}{l_1 + l_2} & s_1 = s_2 \end{cases} \end{cases} \tag{B.6}$$

- Finding the "better edge" b_e: the edge where the 1st digit of your score is greatest. In case of equality between 2 edges, which is that the 2nd digit of this score is greatest.
- Finding the "better node" b_n: this is the end node whose 1st digit of your score is greatest. In the case of equality, the node that 2nd digit of the score is the smallest (the node closest to the interface). The node b_n is the single node selected in this iteration:

$$S_n^k = \{b_n\} \tag{B.7}$$

The algorithm stops if during an iteration the whole set S_e^k becomes empty. The final set of selected nodes will be:

$$W = \bigcup_k S_n^k \tag{B.8}$$

After this selection phase of the nodes, the algorithm constructs the space of Lagrange multipliers, which size is equal to that W. Thus, the multipliers space is:

$$S_\lambda = \left\{ \lambda^i, i \in \left\{ 1, setting\left(W\right) \right\} \right\} \tag{B.9}$$

All the edges e have a common edge $i \in W$ given Lagrange multipliers λ_e equal among them:

$$\forall e \in S_e, \quad \forall i \in W, \quad \lambda_e = \lambda^i \qquad \text{if} \quad v_1\left(e\right) = i \quad \oplus \quad v_2\left(e\right) = i \tag{B.10}$$

where the symbol $\oplus$ refers to the "or exclusive".

Edges with both ends in W have Lagrange multipliers which are linear combinations of multipliers closest. In the original version of the algorithm presented in 2D, a Lagrange multiplier on that edge is a linear combination of Lagrange multipliers and following the previous interface. In 3D, this concept of "precedent" and "after" no longer exists because the concept of "path" along the interface is lost. To define a general linear relationship (in 2D/3D), we use the concept of distance, and each end of the edge on the research Lagrange multiplier nearest covered by an edge connected to e.

B.2 Description of the modified algorithm (algo2)

The algorithm is based on the work of Geniaut, Massin, Moës [49]. Based on similar ideas, a new algorithm was proposed, which seeks to focus on linear relations to relations of equality between Lagrange multipliers. Thus, in the new version, we consider all edges on which the normal zero level set at least one point. Those edges connecting these points on both sides of the interface (or any points on the interface). The algorithm seeks the minimum subset of edges to connect all the ends of edges. Then, groups of connected edges are extracted. Relations are imposed as follows:

- Multipliers on the edges of each group are imposed equal,
- The multipliers on the remaining edges are linear combinations of other multipliers.

In more formally, let E and N are the sets containing all the edges and all nodes of the mesh. The two ends of an edge $e \in E$ are noted $\left(v_1\left(e\right), v_2\left(e\right)\right) \in N^2$. It first determines S_e, the set of edges that were severely cut by the interface. The interface is represented by the normal level set lsn, an edge $e \in E$ is severely cut by the interface if and only if $lsn\left(v_1\left(e\right)\right) \cdot lsn\left(v_2\left(e\right)\right) < 0$. Note that if the interface coincides with the node $v_1\left(e\right)$ or node $v_2\left(e\right)$, the edge e is not in S_e:

$$S_e = \left\{ \mathrm{e} \in E, \quad lsn\left(v_1\left(e\right)\right) \cdot lsn\left(v_2\left(e\right)\right) < 0 \right\} \tag{B.11}$$

Considered N_e the set of nodes connected by the elements S_e. It separates N_e into two parts: nodes "below" and "above" the crack, as the sign of lsn:

$$\begin{aligned} &N_e = \left\{ n \in N, \quad \exists e \in S_e \quad n = v_1\left(e\right) \quad \text{or} \quad n = v_2\left(e\right) \right\} \\ &N_e^+ = \left\{ n \in N_e, \quad lsn\left(n\right) > 0 \right\} \quad and \quad N_e^- = \left\{ n \in N_e, \quad lsn\left(n\right) < 0 \right\} \end{aligned} \tag{B.12}$$

The search algorithm S_{ve}, the minimum subset of S_e that connects the nodes in N_e^+ and the nodes in N_e^-. Each node in N_e^+ must be connected to at least one node in N_e^-, and each node in N_e^- must be connected to at least one node N_e^+. The edges in S_{ve} are called "critical edges", because if one of these edges disappears, at least one node in N_e is an orphan. This set of critical edges is not necessarily unique. In the presence of selection, the most vital edge short is preferred. As it will be shown, this amounts to minimize the approximation P0. For research of all S_{ve}, we chose an algorithm based on the concepts of scores of nodes and edges, a concept that is found in the algo1. The algorithm will remove one to all non-essential edges, until there is no more edges as vital. More precisely, we associate a score to each node, which corresponds to the number of edges connected to that node. At each edge, we associate a score, which is the minimum scores on both ends nodes. Let e the edge with the highest score (with the same score, the longest edge is preferred). If the score of e is equal 1, then all the edges are still vital edges. S_{ve} is determined and the algorithm stops. If the score of e is strictly greater than 1, the edge e is an edge non-vital and symbolically removed from the list of edges S_e. The algorithm starts again with a new calculation of the score of the nodes, and so on until there is no more edges as vital.

It is important to note that S_{ve} consists of some edges disconnected, and some certain edges connected between them. These groups of edges are connected vital extracts the S_{ve}. In this group, all edges are connected with a single node (see Figure B.2.1). Let G_{cve}^i the group vital edges connected by the node i. So G_{cve}^i is defined by:

$$G_{cve}^i = \left\{ e \in S_{ve}, \quad i = v_1(e) \quad \text{or} \quad i = v_2(e) \right\} \tag{B.13}$$

All multipliers carried by edges in the same group are imposed equal. The other multipliers are carried by non-critical edges. These multipliers are not essential for the approximation of the contact pressure. They are imposed thus as linear combinations of multipliers on edges vital. These linear combinations are determined by the following procedure. Let λ_e, $e \in S_e \setminus S_{ve}$ the Lagrange multiplier given by a non-vital edge e. For each end of the edge e, we search the closest (in terms of physical distance, not the distance between centre nodes) λ_k increased by an edge connected to e:

$$for\ i = 1,2 \quad find\ k_i = \arg\min_k \left\{ \text{dist}\left(\lambda_e, \lambda_k\right), k \in G_{cve}^{v_i(e)} \right\} \tag{B.14}$$

The linear relationship is imposed between λ_e, λ_{k_1} and λ_{k_2}:

$$\lambda_e = \frac{\text{dist}\left(\lambda_e, \lambda_{k_2}\right)}{\text{dist}\left(\lambda_e, \lambda_{k_1}\right) + \text{dist}\left(\lambda_e, \lambda_{k_2}\right)} \lambda_{k_1} + \frac{\text{dist}\left(\lambda_e, \lambda_{k_1}\right)}{\text{dist}\left(\lambda_e, \lambda_{k_1}\right) + \text{dist}\left(\lambda_e, \lambda_{k_2}\right)} \lambda_{k_2} \tag{B.15}$$

To illustrate this algorithm, consider the 2D case in Figure B.2.1. These edges cut by the interface are:

$$S_e = \{1-8\,;\,2-8\,;\,2-9\,;\,2-10\,;\,3-10\,;\,3-11\,;\,4-12\,;\,5-12\,;\,6-12\,;\,7-12\,;\,7-13\} \tag{B.16}$$

The set of critical edges (drawn as a solid line in Figure B.2.1) and non-critical edges (in dashed line) are as follows:

$$\begin{aligned} S_{ve} &= \{1-8\,;\,2-9\,;\,2-10\,;\,3-11\,;\,4-12\,;\,5-12\,;\,6-12\,;\,7-13\} \\ S_e \setminus S_{ve} &= \{2-8\,;\,3-10\,;\,7-12\} \end{aligned} \tag{B.17}$$

It may be noted that the edge $\{2-10\}$ or the edge $\{3-10\}$ may be chosen either as vital edge. But as mentioned before, the shortest edge is preferred, therefore $\{2-10\} \in S_{ve}$ and not $\{3-10\}$. Two groups of connected critical edges can be removed (they are circled in dotted lines in Figure B.2.1):

$$G^2_{cve} = \{2-9\,;\,2-10\} \quad \textit{and} \quad G^{12}_{cve} = \{4-12\,;\,5-12\,;\,6-12\} \tag{B.18}$$

For each group, the Lagrange multipliers are imposed equal:

$$\lambda_C = \lambda_D \quad \textit{and} \quad \lambda_G = \lambda_H = \lambda_I \tag{B.19}$$

For each non-critical edge $\{2-8\}$, $\{3-10\}$ and $\{7-12\}$, and a linear relationship between the multiplier is imposed:

$$\lambda_B = \frac{BC \cdot \lambda_A + AB \cdot \lambda_C}{AB + BC}, \quad \lambda_E = \frac{EF \cdot \lambda_D + DE \cdot \lambda_F}{DE + EF}, \quad \lambda_J = \frac{JK \cdot \lambda_I + IJ \cdot \lambda_K}{IJ + JK} \tag{B.20}$$

In view of relations imposed on the Lagrange multipliers, one can determine the degree of approximation of contact pressures along the interface. For pieces of interface where the multipliers are equal, the approximation is P0. And the pieces of the interface where multipliers are linear combinations, the approximation is P1. The Figure B.2.1 compares resulting approximations between the two algorithms.

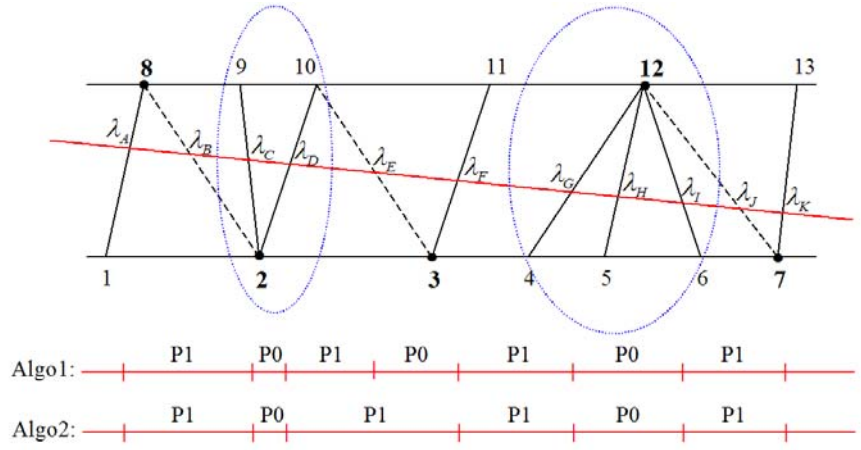


Figure B.2.1. *Example of edges cut by an interface and approximation results.*

- Select the essential edges: these are the minimum set of edges connecting the nodes from one side to the other.
- Attach a Lagrange multiplier to each essential edge.
- If a set of edges emanate from the same node, they share the same Lagrange multiplier.
- For the non essential edges, the Lagrange multiplier is obtained by linear combination.

Remarks on relations imposed by the algorithm 1 or 2 for the multipliers of contact

It is the linear relationship between the Lagrange multipliers of contact λ_1, λ_2 and λ_3:

$$\lambda_2 = \alpha\lambda_1 + (1-\alpha)\lambda_3 \tag{B.21}$$

The relationship focuses on the pressure and not on the vector contact pressure. In case of a curved structure the relationship is on the vector of pressure type:

$$\lambda_2\mathbf{n_2} = \alpha\lambda_1\mathbf{n_1} + (1-\alpha)\lambda_3\mathbf{n_3} \tag{B.22}$$

is not possible because the vector $\mathbf{n_2}$ is not an unknown.

B.3 Algorithm to define the mesh on the interface

The algorithm was introduced by Béchet, N. Moës, B. Wohlmuth 2008 [8]. The interface Γ is cutting through edges of the two-dimensional mesh T_h defining a graph. The vertices of this graph are vertices of the mesh T_h which are located exactly on Γ or are intersection nodes of an open edge of the mesh T_h and Γ. Once all the vertices of the graph have been marked, we connect the vertices based on the following rule: two vertices are connected in the graph if both of them result from the intersection of an open edge with Γ and the associated two edges share an endpoint. We note that vertices being vertices of the original mesh are always isolated vertices. In Figure B.3.1, Γ is indicated by a dashed line. On the left, the mesh is aligned with the interface, and the graph is a set of isolated vertices. On the right, the interface crosses elements but the vertices are not connected because they do not lie on edges sharing common nodes. The situation is different in Figure B.3.2 here vertices of the graph are connected.

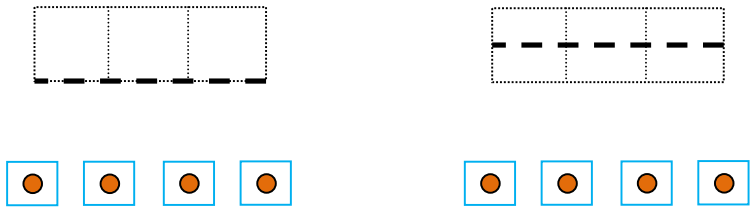


Figure B.3.1. *A mesh conforming to* Γ *(left) and not conforming (right). Below each case is the vertex graph as well as the selected vital vertices (squared).*

The naive approach for building the Lagrange multiplier space is to set one independent multiplier on each vertex of the graph. We know that this does not pass the numerical inf-sup test (Ji and Dolbow 2004 [15]).

We thus choose to select a subset of the vertices in the graph to define independent Lagrange multipliers. We call them vital vertices. They are selected based on the following rules

(i) An isolated vertex is always vital.
(ii) A vital vertex is not allowed to be connected to any other vital vertex.
(iii) A non vital vertex must be connected to at least one vital vertex.

The squared vertices in Figure B.3.1 and B.3.2 are vital. Note that for a given graph, the choice of vital vertices is not unique, see Figure B.3.2.

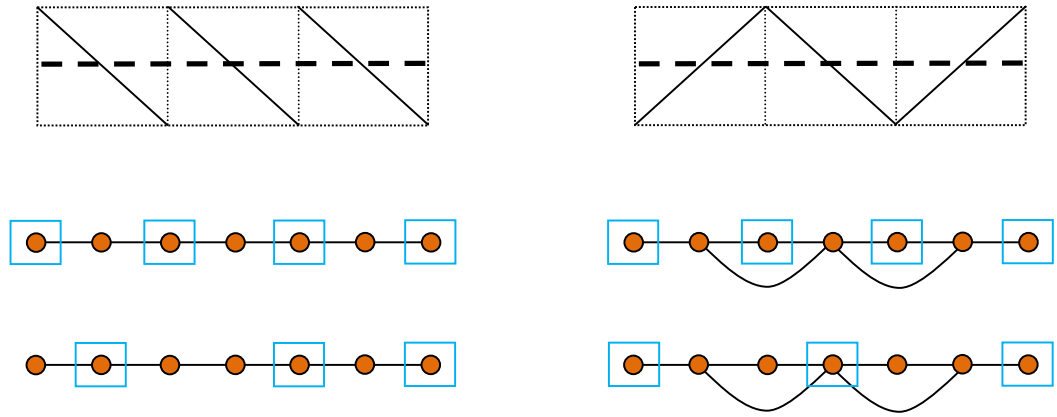


Figure B.3.2. *Two different situations of a non-adapted mesh. Below each case is the vertex graph as well as possible vital vertices.*

The algorithm for the selection of the vital vertices that is used in the implementation closely follows the rules defined above. It can handle 3D problems with surface boundary conditions as well.

Algorithm in 2D and 3D used to define the vital vertices.

0. Define an empty set of vital vertices called *vital* and an empty set of regular (non vital) vertices called *non_vital*.
1. The vertices on the interface that are also vertices of the mesh are introduced in the vital set. The rest of the vertices on the interface, denoted V, emanate from cut edges. For each such vertex $v_i \in V$, we denote by $v_i[k]$; k = 0, 1., the end-points of the cutting edge.
2. For every vertex $v_i \in V$ on the interface, count the number of intersections of the interface by the edges incident to $v_i[k]$. This defines the set $n_{\text{int}}[v_i]$.
3. Sort the set $n_{\text{int}}[v_i]$ (low number of intersections first).
4. Loop: Pick up the first item of $n_{\text{int}}[v_i]$ and the corresponding v_i.
5. Check that for the every node $v_i[k]$, none of its incident edges are intersecting the interface at an already vital vertex. If this condition is fulfilled, mark v_i as *vital* and also mark every vertex that is the intersection of the interface from an edge incident to $v_i[k]$ as *non_vital*. If the condition does not hold, simply mark v_i as *non_vital*.

6. Remove v_i from $n_{\text{int}}[v_i]$.
7. If $n_{\text{int}}[v_i]$ is not empty, go to 4.
8. The set called *vital* contains the vital vertices. The set called *non_vital* contains the other vertices.

REMARK B.3.1 Roughly speaking criteria (iii) makes the set of vital vertices as large as possible to get a good best approximation property for the discrete Lagrange multiplier space, and criteria (ii) makes the dimension of L^h as small as necessary to satisfy a uniform inf-sup condition. Criteria (i) guarantees that in the case of an aligned interface, the one-dimensional mesh is inherited from the two-dimensional mesh.

B.4 Definition of a stable Lagrange multiplier space

The details on this matter may be found in Béchet, Moës and Wohlmuth [8]. For each vital vertex $p \in V_h$, we define the associated basis function $\mu_p \in L^h$ as a linear combination of some nodal hat functions ϕ_q, $q \in P_h$ restricted to Γ. The definition of the coefficients is based on some preliminary observations and remarks. We note that the number of vertices in P_h such that $\phi_q|_\Gamma$ is not equal to zero is of order h^{-1}. Introducing the subset P_h^Γ, see Figure B.4.1, by

$$P_h^\Gamma := \left\{ p \in P_h ; \phi_p|_\Gamma \text{ not identical zero} \right\},$$

it is trivial to see that $\sum_{q\in P_h} \alpha_{pq}\phi_q|_\Gamma = \sum_{q\in P_h^\Gamma} \alpha_{pq}\phi_q|_\Gamma$, and thus we set

$$\mu_p := \sum_{q\in P_h^\Gamma} \alpha_{pq}\phi_q|_\Gamma, \quad p \in V_h.$$

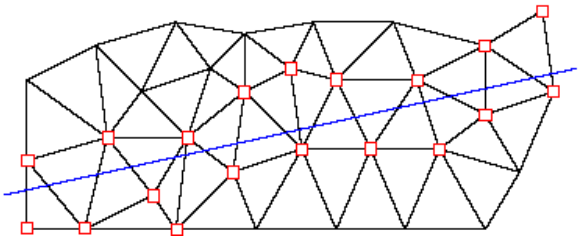


Figure B.4.1. *Set of vertices in* $\wp_h^\Gamma$.

In a next step, we define for each vital vertex $q \in V_h$ a $P_q \in P_h^\Gamma$. We recall that each vital vertex q is in P_h or is the intersection point of an open edge e_q of the mesh T_h. In the first case, we set $P_q := \{q\}$ and in the second case, we define $P_q := \left\{ p_q^1, p_q^2 \right\}$, where p_q^1, p_q^2 are the two endpoints of e_q. In addition, we set

$$\mathbb{Q}_h^\Gamma := P_h^\Gamma \setminus U_{q\in V_h} P_q =: P_h^\Gamma \setminus P_h^V$$

and observe that $\mathbb{Q}_h^\Gamma$ might be the empty set. Due to criteria (iii), each $q \in \mathbb{Q}_h^\Gamma$ is connected by at least one closed edge $\overline{e}_q$ cutting the interface with an element in P_h^V, see also Figure B.4.2. The number of such edges e_q, i.e., one endpoint is *q*, the other endpoint q_0 is in P_h^V and $\overline{e}_q \bigcap \Gamma$ is not empty, is denoted by n_q. Because all P_p, $p \in V_h$, are pairwise disjoint, there exists a unique $p \in V_h$ such that $q_0 \in P_p$,and we put q into $\mathbb{Q}_p$,i.e., $U_{p \in V_h} \mathbb{Q}_p = \mathbb{Q}_h^\Gamma$.

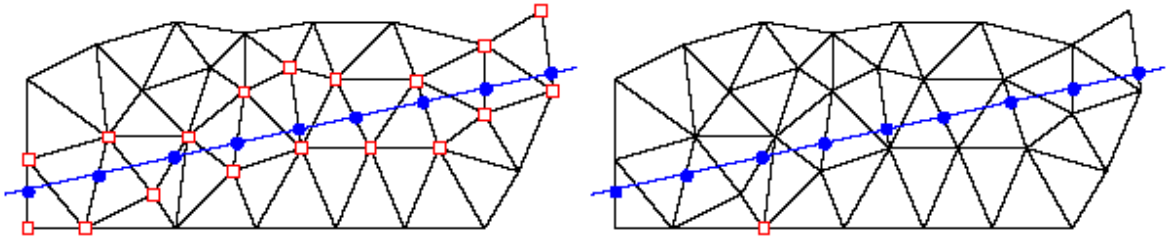


Figure B.4.2. *The set of vertices in $\wp_h^\Gamma$ (left) and in $\mathbb{Q}_h^\Gamma$ (right) are marked with empty squares, the vertices in V_h are marked with filled circles.*

In the example of the right picture in Figure B.4.3, $n_q = 1$ or $n_q = 2$ .We note that $\mathbb{Q}_p$ can be empty and that $\mathbb{Q}_p \bigcap \mathbb{Q}_q$ does not have to be empty for $p, q \in V^h$, see the right picture in Figure B.4.3.

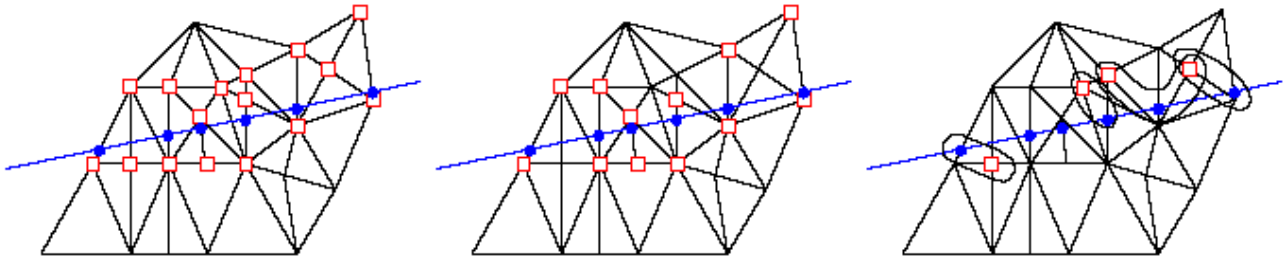


Figure B.4.3. *Sets of vertices from the left to the right $\wp_h^\Gamma$, $\wp_h^v$, $\mathbb{Q}_h^\Gamma$ and Q_p.*

In terms of these subsets, we define now the values of the coefficients $\alpha_{pq}, p \in V_h, q \in P_h^\Gamma$

$$\alpha_{pq} := \begin{cases} 1, & q \in P_p \\ \dfrac{1}{n_p}, & q \in \mathbb{Q}_p \\ 0, & otherwise \end{cases}$$

Then μ_p has a local support and can be written as

$$\mu_p = \sum_{q \in P_p} \phi_q + \sum_{q \in \mathbb{Q}_p} \frac{1}{n_q} \phi_q$$

Figure B.4.4 shows which nodes $q \in \wp_h^\Gamma$ contributes to the definition of μ_p, $p \in V_h$

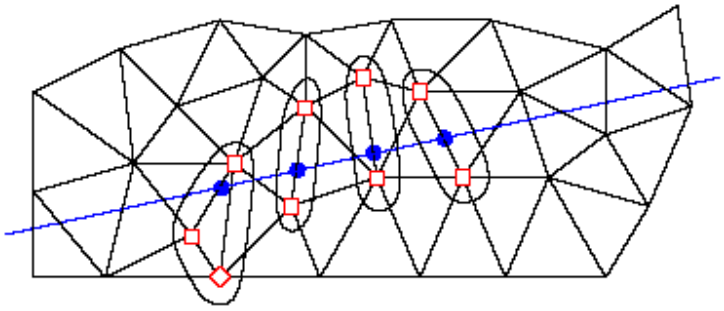


Figure B.4.4. *Nodes in $\wp_p$ are marked with a square, nodes in $\mathbb{Q}_p$ with a diamond.*

LEMMA B.4.1 The set $\{\mu_p\}_{p \in V_h}$ forms a positive partition of unity with local supports on Γ, i.e.,

$$\sum_{p \in V_h} \mu_p = 1$$

REMARK B.4.2 Lemma B.4.1 yields that the Lagrange multiplier space L^h reproduces constants and thus the best approximation property gives an $O(h)$ term in the a priori analysis.

LEMMA B.4.3 There exist constants independent of the mesh-size such that for all elements $e \in \varepsilon_h$

$$ch_T \le h_e \le Ch_T, \quad T \in T_e$$

where h_e is the length of e, and h_T is the diameter of the element T, and $T_e \subset T_h$ is the set of all elements T such that the intersection with e is not empty.

THEOREM B.4.4 There exists a constant independent of the mesh-size such that for all $\mu^h \subset L^h$

$$\sup_{v^h \in U^h} \frac{\int_\Gamma \mu^h \left[v^h \right] d\Gamma}{\left\| v^h \right\|_{1;\Omega}} \ge c \left\| \mu^h \right\|_{-\frac{1}{2};\Gamma_h}$$

REMARK B.4.5 Both algorithms, the one to construct the vital vertices and the one to define the basis functions of L^h are not restricted to the two-dimensional setting. They can easily be generalized to the three-dimensional case. Using the alternative approach of using standard hat functions for the Lagrange multiplier on the mesh ε_h would require in 3D to construct a mesh of the interface from the vital vertices, that satisfies some regularity requirements (e.g. using a Delaunay approach). By using for the definition of the Lagrange multiplier shape functions defined on the background mesh this can be avoided.