# Adaptive $n$-sided Polygonal Finite Element Method for analysis of plane problems.

Balaji K S

A Thesis Submitted to

Indian Institute of Technology Hyderabad

In Partial Fulfillment of the Requirements for

The Degree of Master of Technology

भारतीय प्रौद्योगिकी संस्थान हैदराबाद
Indian Institute of Technology Hyderabad

Department of Civil Engineering

June 2013

# Declaration

I declare that this written submission represents my ideas in my own words, and where ideas or words of others have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the Institute and can also evoke penal action from the sources that have thus not been properly cited, or from whom proper permission has not been taken when needed.

——————————

(Signature)

——————————

(Balaji K S)

——————————

(Roll No.)

# Approval Sheet

This Thesis entitled Adaptive $n$-sided Polygonal Finite Element Method for analysis of plane problems. by Balaji K S is approved for the degree of Master of Technology from IIT Hyderabad

———————————

( Dr. Manoj Pandey ) Examiner
Dept. of Mech Eng
IITM

———————————

(Dr. KVL Subramaniam) Examiner
Dept. of Civil Eng
IITH

———————————

(Amirtham Rajagopal) Adviser
Dept. of Civil Eng
IITH

———————————

(Amirtham Rajagopal) Chairman
Dept. of Civil Eng
IITH

# Acknowledgements

I would like to convey my special thanks to my thesis advisor Dr. Amirtham Rajagopal for all his faith and confidence in me. He believed in me and has given me, such an awesome topic to work on it. Working with him and being a part of his research group has been a privilege for me. It would have been highly impossible for me to move forward without his motivation and well organized guidance. His encouragement and technical guidance made my thesis work worth full and interesting. Working with him was indeed a fantastic, fruitful, and an unforgettable experience of my life.

I am also indebted to say my heartily thanks to my Head of Department, Prof. K.V.L.Subraminam for providing a good environment for working. It gives me immense pleasure to thank my faculty members who taught during my M.Tech course work. I would like to thank Prof. M.R. Madhav, Dr. B. Umashakar, Dr. Shashidhar and Dr. K.B.V.N.Phanindra, Dr. Mahendrakumar Madhavan for their valuable suggestion during my course work.

I am thankful to my prestigious institute Indian Institute of Technology, Hyderabad for all the support and providing everything whatever was required for my research work.

I am grateful and thankful all my friends. I would like to say special thanks to my supervisor's research group members Nitin (M.Tech.), Mahendra Kumar Pal (M.Tech.), L. Harish (M. Tech.), B.Umesh (PhD.), Balakrishana (PhD.). I also thank all my friends, Basket ball team, for making my stay wonderful at IIT Hyderabad.

## Dedication

# To My Parents

# Abstract

In this work we present an adaptive polygonal finite element method for analysis of two dimensional plane problems. The generation of $n-$ sided polygonal finite element mesh is based on generation of a centroidal Vorononi tessellation (CVT). By this method an unstructured tessellation of a scattered point set, that minimally covers the proximal space around each point in the point set can be generated. The method has also been extended to include tessellation for non convex domains. For the numerical integration of Galerkin weak form over polygonal finite element domains we resort to classical Gaussian quadrature applied on triangular sub domains of each polygonal element. An adaptive finite element analysis strategy is proposed and implemented in the present work. A patch recovery type of stress smoothing technique that utilizes polygonal element patches for obtaining smooth stresses has been proposed for obtaining the smoothed finite element stresses. A classical $z^2$ type $a - posteriori$ error estimator that estimates the energy norm of the error from the recovered solution is then implemented. The refinement of the polygonal elements is made on an element by element basis through a refinement index. Numerical examples of two dimensional plane problems are presented to demonstrate the efficiency of the proposed adaptive polygonal finite element method.

# Contents

# Chapter 1

# Introduction

## 1.1 Introduction

Finite element method is a powerful numerical tool for solving partial differential equations. The use of two dimensional triangular, quadrilateral elements and three dimensional tentrahedral, hexahedral elements has been popular. However, there are associated complexities such as developing robust and fast algorithms for generating quality meshes on two or three dimensional complex geometries of microstructures, distortion effects under large deformation, complexities in development and use of higher order elements and need of efficient quadrature schemes for evalutation of integrals amongst others. The use of polygonal elements with $n-$ number of sides will provide greater flexibility and better accuracy to address some of these problems. Polygonal finite element discretization can be useful in many areas like, the nonlinear constitutive modeling of polycrystalline materials with general anisotropic behavior or ferroelectrics [1, 2] where each grain with its independent properties is represented by one element, for interface elements for connecting dissimilar finite element meshes [3], for two field methods for solving diffusion equations [4], for solid mechanics problems [5] including incompressible materials [6], and for topology optimization [7]. The recent focus in polygonal FEM has also been on generating conformal polygonal discretizations [8–10], developing numerical integration schemes for polygonal finite element methods either based on conformal mapping [11,12] or generalizing Gaussian quadrature rules [13,14], and the natural element method for non-convex and discontinuous problems [15].

The quality of finite element solution is not equally good in all regions of the domain under consideration, thus making a locally finer mesh necessary. The general recognition that a reliable numerical approximation of the analytical solution of a given mathematical model needs the notion of discretization errors with bounds. Initial contribution in this regard can be attributed to Babuska et al [16] and the other works by Babuska and Rheinboldt [17], presenting a priori error estimators in the energy norm for second-order adjoint elliptic boundary value problems. The breakthrough for error-controlled adaptivity of primal finite element methods (the displacement approach) was achieved by Babuska and Miller [18], presenting the a posteriori error analysis of the discretization error for second-order elliptic problems in the global energy norm with bounds from below and from above. At the same time Zienkiewicz and Zhu [19] presented a heuristically motivated $a-posteriori$ averaging based error indicator using explicitly calculated smoothened stresses, correctly assuming that the smoothened stresses are more accurate than the discretized ones. This indicator is easy to implement into existing finite element programs, and has been used as as basis for advanced adaptive strategies in some of the recent works [20–23]. A more detailed review on advances in error estimation strategies can be obtained in [24] and [25]. There has been recent works on adaptive singular finite element [26] and refinement strategies based on asymptotic expansions of finite element solution [27].

In this work we present an adaptive polygonal finite element method for analysis of two dimensional plane problems. The generation of $n-$ sided polygonal finite element mesh is based on procedure of centroidal Vorononi tessellation (CVT). By this method an unstructured tessellation of a scattered point set that minimally covers the proximal space around each point can be generated. The method has also been extended to include non convex domains. For the numerical integration of Galerkin weak form over polygonal finite element domains we resort to classical Gaussian quadrature applied on triangular sub domains of each polygonal element. An adaptive refinement strategy is proposed in the present work. A patch recovery type of stress smoothing technique that utilizes polygonal patches for obtaining smooth stresses has been proposed for obtaining the smooth stresses. A classical $z^2$ type $a-posteriori$ error estimator that estimates the energy norm of the error from the recovered solution is then implemented. The refinement of the polygonal elements is made on an element by element basis via a refinement index. Numerical examples of two dimensional plane problems are presented to demonstrate the efficiency of the proposed adaptive polygonal finite element method

The outline of the thesis is as follows : in Chapter 2, after a brief review of mathematical preliminaries we present an overview of generating the polygonal finite element mesh using CVT and details of extension of methodology to non convex domains is presented. In Chapter 3 we discuss the non polynomial type polygonal interpolant used in the present work. Chapter 4 presents equilibrium equations, weak form and discretized weak form for two dimensional plane elasticity problems. A numerical integration strategy performed by dividing the polygonal domain into triangular domain and then using well know quadrature rules on a triangle is presented in Chapter 5. Chapter 6 presents the details of *a-posteriori* error estimates and adaptive refinement. In the last Chapter 7, we present numerical examples of plane problems and study the convergence characteristics under uniform and adaptive mesh refinement.

# Chapter 2

# Mesh Generation

In this chapter we present a simple and robust method for polygonal mesh generation that relies on implicit description of domain geometry. The main components of mesh generator are implicit description of the domain and the centroidal Voronoi tessellation (CVT) [28] used for its discretization. The signed distance function contains all the essential information about the domain geometry and offers great flexibility to construct relatively large class of domains. The Loyd's method is used to ensure uniform (optimal) distribution of seeds and thus a high quality mesh (Talischi et al. [29]). Examples are provided to illustrate capabilities of the proposed methodology.

## 2.1 Mathematical preliminaries

### Distance function and implicit representation

Let $\Omega$ be a subset of $\mathbb{R}^2$. The *signed distance function* associated with $\Omega$ is defined by

$$d_\Omega(\boldsymbol{X}) = S_\Omega(\boldsymbol{X}) \min_{\boldsymbol{Y} \in \partial\Omega} \|\boldsymbol{X} - \boldsymbol{Y}\| \tag{2.1}$$

where $\partial\Omega$ denotes the boundary of $\Omega$. $\|\cdot\|$ is the standard Euclidean norm in $\mathbb{R}^2$. Here $\|\boldsymbol{X} - \boldsymbol{Y}\|$ is the distance between any point $\boldsymbol{X}$ and point $\boldsymbol{Y}$ on the boundary of domain, and the sign function is given by

$$S_\Omega(\boldsymbol{X}) \colon = \begin{cases} -1, & \text{if } \boldsymbol{X} \in \Omega \\ +1, & \text{if } \boldsymbol{X} \in \mathbb{R}^2 \setminus \Omega \end{cases}$$

Thus, if $\boldsymbol{X}$ lies inside the domain $\Omega$, $d_\Omega(\boldsymbol{X})$ is minus the distance of $\boldsymbol{X}$ to the closest boundary point. The following characterization are made from this definition:

$$\Omega = \left\{\boldsymbol{X} \in \mathbb{R}^2 \ : \ d_\Omega(\boldsymbol{X}) \leq \ 0\right\} \tag{2.2}$$

$$\partial\Omega = \left\{\boldsymbol{X} \in \mathbb{R}^2 \ : \ d_\Omega(\boldsymbol{X}) = 0\right\} \tag{2.3}$$

For the domain shown in Fig.2.1, consider two points $\boldsymbol{A}$ and $\boldsymbol{Z}$ which are outside and inside the

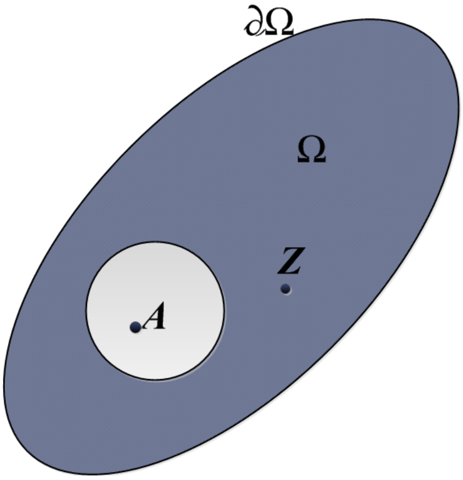


Figure 2.1: For the domain above $d_\Omega(\boldsymbol{Z}) < 0$ and $d_\Omega(\boldsymbol{A}) > 0$.

domain Ω, respectively. Thus distance function of point $\boldsymbol{A}$ is positive, $d_\Omega(\boldsymbol{A}) > 0$, and that of $\boldsymbol{Z}$ is negative, $d_\Omega(\boldsymbol{Z}) < 0$, implying that points $\boldsymbol{A}$ and $\boldsymbol{Z}$ are outside and inside the domain Ω, respectively.
We can see from the discussion so far that when Ω is characterized by its signed distance function, a greater information about Ω can be readily extracted. The essential task then is to construct $d_\Omega(\boldsymbol{X})$ for a given domain Ω that we wish to discretize. For example, if Ω is a circle of radius $r$ centered at point $\mathbf{X}_0$, (See Fig.2.2) the distance function for that domain is given by,

$$d_\Omega(\boldsymbol{X}) = \|\boldsymbol{X} - \boldsymbol{X}_o\| - r \tag{2.4}$$

Other useful information about domain geometry given by signed distance function is direction of

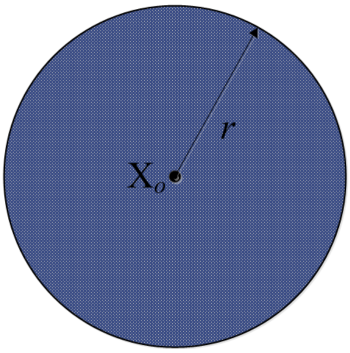


Figure 2.2: A circular domain of radius $r$.

nearest boundary point, given by its gradient $\boldsymbol{\nabla} d_\Omega(\boldsymbol{X})$. We use the property of gradient to find reflection of any point $\boldsymbol{X}$ about the closest boundary point. We denote reflection by $R_\Omega(\boldsymbol{X})$, see Fig.3.

$$R_\Omega(\boldsymbol{X}) = \boldsymbol{X} - 2d_\Omega(\boldsymbol{X})\boldsymbol{\nabla} d_\Omega(\boldsymbol{X}) \tag{2.5}$$

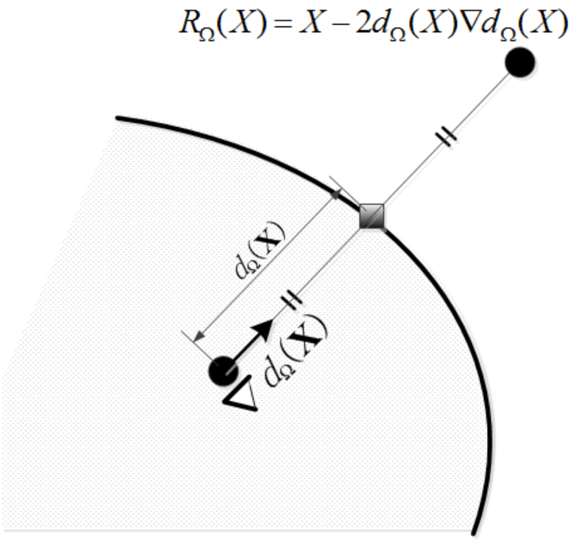

$R_\Omega(X) = X - 2d_\Omega(X)\nabla d_\Omega(X)$
$d_\Omega(\boldsymbol{X})$
$\nabla d_\Omega(\boldsymbol{X})$

Figure 2.3: Reflection of a point $\boldsymbol{X}$ about nearest boundary point $\boldsymbol{X}_b$

This reflection property is used in mesh generation to obtain a polygonal mesh confining to defined geometry (Explained in Sec.2.3).

## 2.2 Centroidal Voronoi tessellation (CVT) and Lolyd's algorithm

Consider a set of nodes $\mathcal{P} = \{\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots \boldsymbol{P}_M\}$ with $\boldsymbol{P}_I \in \mathbb{R}^2$. The first order Voronoi diagram $\mathcal{V}(\mathcal{P})$ of the set $\mathcal{P}$ is a subdivision of the Euclidean space $\mathbb{R}^2$ into convex regions

$$\mathcal{V}(\boldsymbol{P}_I) = \left\{\boldsymbol{x} \in \mathbb{R}^2 : ||\boldsymbol{P} - \boldsymbol{P}_I|| < ||\boldsymbol{P} - \boldsymbol{P}_J|| \;\forall\; J \neq I\right\}$$

called Voronoi cells, where $\mathcal{V}(\mathcal{P}) = \cup\mathcal{V}(\boldsymbol{P}_I)$, as shown in Fig.2.4.
The above definition states that any point $\boldsymbol{x}$ in the Voronoi cell $\mathcal{V}(\boldsymbol{P})$ is closer to node $\boldsymbol{P}_I$ than to any other node $\boldsymbol{P}_J$. The properties of Voronoi diagrams have been studied extensively and we refer the reader to review paper (Aurenhammer 1991) on this topic.

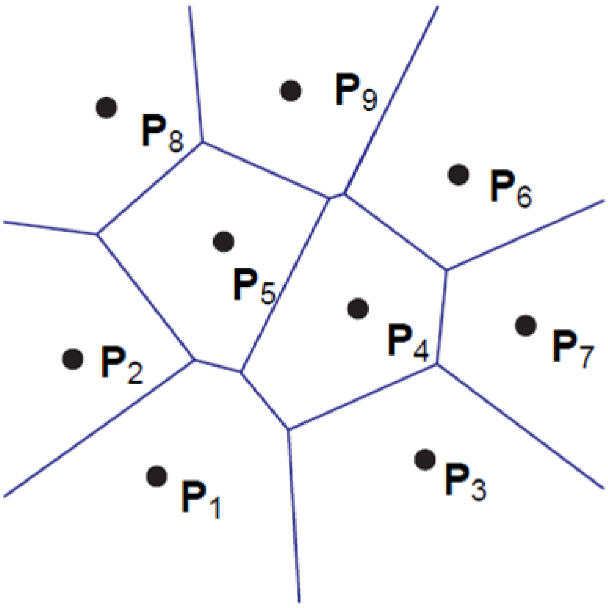


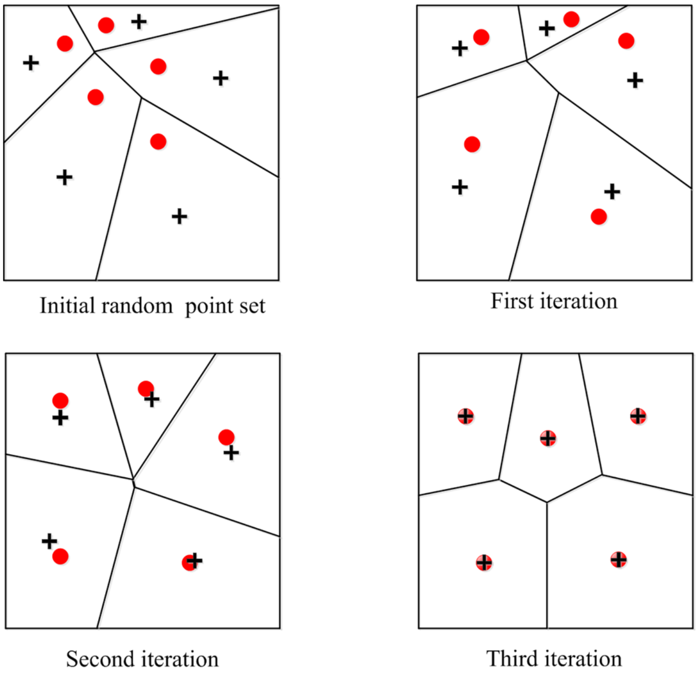
Figure 2.4: voronoi diagram of given point set.

The regularity of Voronoi diagram is determined entirely by the distribution of generating point set. A random set of generators may lead to a discretization not suitable for use in finite element analysis. Therefore, we restrict our attention to special class of Voronoi tessellation that enjoy higher level of regularity, centroidal Voronoi tessellation (CVT). A Voronoi tessellation is *centroidal* if for every $\boldsymbol{P}_I \in \mathcal{P}$, $\boldsymbol{P}_I = \boldsymbol{P}_c$ where,

$$\boldsymbol{P}_c = \frac{\int_{\mathcal{V}(\boldsymbol{P}_I)} \boldsymbol{x}\rho(\boldsymbol{x})d\boldsymbol{x}}{\int_{\mathcal{V}(\boldsymbol{P}_I)} \rho(\boldsymbol{x})d\boldsymbol{x}} \tag{2.6}$$

and $\rho(\boldsymbol{x})$ is a given density function defined over $\mathcal{V}(\boldsymbol{P}_I)$. Hence in centroidal Voronoi tessellation or Voronoi diagram, the generating point of each Voronoi cell is also the centroid (center of mass) of corresponding Voronoi region. It can be viewed as an optimal partition corresponding to an optimal distribution of generators. A number of algorithms are available to generate centroidal Voronoi tessellation. We adopt simple and powerful *Lolyd's algoritm* for generating CVTs.

*Lolyd's algoritm* also known as Voronoi iteration, is a deterministic algorithm for generating CVTs. It is a iterative process between computing Voronoi diagrams and mass centroids. It starts with an initial distribution of point set which iteratively replaces the given point set by centroids of corresponding Voronoi regions. Fig.2.5 and algorithm 1 explains the procedure.



Initial random point set

First iteration

Second iteration

Third iteration

Figure 2.5: The Voronoi diagram of the current points ● at each iteration is shown. The + sign denotes the centroids of the Voronoi cells.

**Algorithm 1** Lolyd's method

```
1: Initialization: Select an initial set of $k$ points $\{\boldsymbol{P}_i\}_{i=1}^{k}$, represented as ● in Fig.2.5.
2: Voronoi tessellation: Construct the Voronoi sets $\{\mathcal{V}\}$ associates with $\{\boldsymbol{P}_i\}_{i=1}^{k}$
3: Centroid computation: Determine the mass centroids (Eqn.(2.6)) of the Voronoi sets $\mathcal{V}(\boldsymbol{P}_i)$
   formed in Step 2. These centroids form a new set of points $\{\bar{\boldsymbol{P}}_i\}_{i=1}^{k}$, represented as + sign in
   Fig.2.5.
4: Convergence test: If these new point set meets convergence criterion, terminate, else return
   to Step 2
```

## 2.3 Methodology

Assume $\Omega \subset \mathbb{R}^2$ is a bounded convex domain with smooth boundary and $\mathcal{P}$ is a given set of distinct seeds in $\Omega$. To construct polygonal discretization of $\Omega$, we first reflect all points in $\mathcal{P}$ about the closest boundary point of $\Omega$ and denote resulting set by $R_\Omega(\mathcal{P})$:

$$R_\Omega(\mathcal{P}) = \{R_\Omega(\boldsymbol{P}_I) \colon (\boldsymbol{P}_I) \in (\mathcal{P})\} \tag{2.7}$$

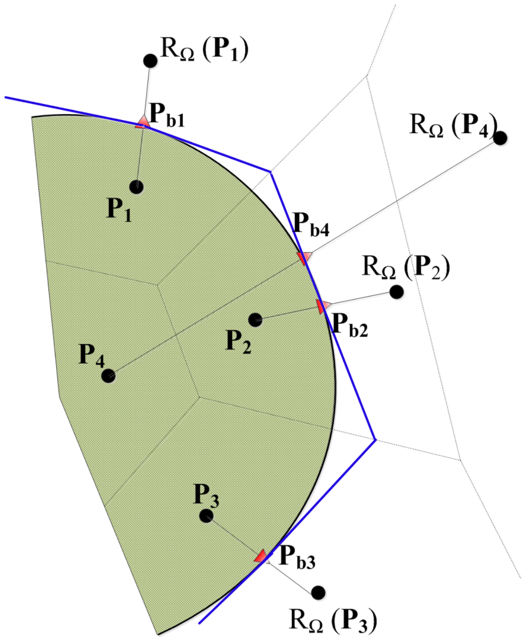


Figure 2.6: Voronoi edges shared between seeds and their reflection approximate boundary of the domain. Reflection of interior seeds, say $\boldsymbol{P}_4$ has no effect on tracing boundary of domain.

Convexity of Ω ensures that all the reflected points lie outside Ω. We then construct Voronoi diagram of original point set as well as its reflection. In other words, we compute $\mathcal{V}(\mathcal{P} \cup R_\Omega(\mathcal{P}))$. If Voroni cells of a point $\boldsymbol{P}_1$ and its reflection have a common edge, i.e., if $\mathcal{V}_{\boldsymbol{P}_1} \cap \mathcal{V}_{R_\Omega(\boldsymbol{P}_1)} \neq \emptyset$, then this edge is tangent to $\partial\Omega$ at $\boldsymbol{P}_{b1}$ as in Fig.6. Therefore, these edges form an approximation to domain boundary and a reasonable discretization of Ω is given by collection of Voronoi cells corresponding to the points in $\mathcal{P}$.

With this basic idea we can extend the procedure for more general domains, in particular those that are non-convex and have piecewise smooth boundaries (presence of corner points on ∂Ω). These features lead to number of complications that require minor modifications of the previous approach. For example, reflecting a point about the nearest boundary point may not be sufficient to capture

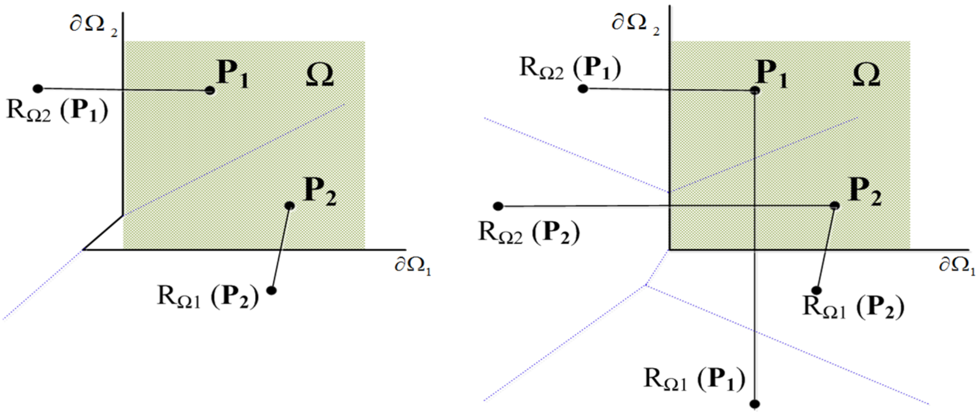


Figure 2.7: To capture corner accurately nearby seeds need to be reflected over both the boundaries approaching corner.

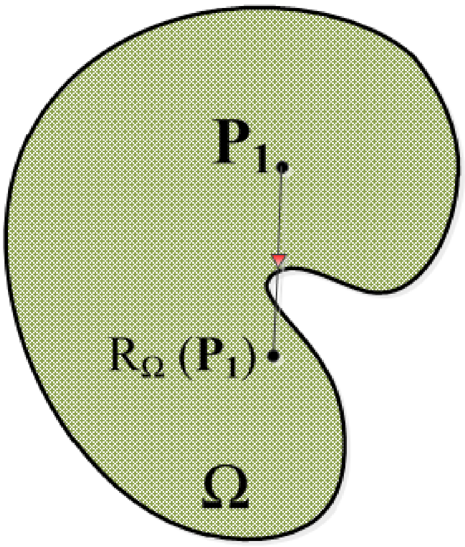


Figure 2.8: In this non-convex domain reflection of point $\boldsymbol{P}_1$ i.e, $R_\Omega(\boldsymbol{P}_1)$ lands inside the domain $\Omega$.

a corner (see Fig.7). We resolve this issue by reflecting seeds about both the boundary segments incident at the corner. Similarly, for non-convex domains, reflection of a seed far from boundary may land inside the domain or interfere with the reflection of another seed, as shown in Fig.8. We check the sign and value of distance function to avoid such scenario. Finally, as seen in Fig.6, the reflection of most of the seeds in the interior of the domain has no effect on approximation of the boundary. Thus we reflect only the seeds that are in a band near to boundary. This significantly reduces the computational cost and improves the robustness of the algorithm.

Any arbitrary geometries can be generated using simple set operations such as union, intersection, and complementation to piece together and combine different geometries. Fig.2.10 shows the various arbitrary domains generated using the code. Consider 2 domains $\Omega_1$ and $\Omega_2$. The sign of distance function for various regions of combined geometry (see Fig.2.9) is given by following expressions.

$$\begin{aligned} d_{\Omega_1 \cup \Omega_2}(\boldsymbol{X}) &= \min\left(d_{\Omega_1}(\boldsymbol{X}), d_{\Omega_2}(\boldsymbol{X})\right) \\ d_{\Omega_1 \cap \Omega_2}(\boldsymbol{X}) &= \max\left(d_{\Omega_1}(\boldsymbol{X}), d_{\Omega_2}(\boldsymbol{X})\right) \\ d_{\mathbb{R}^2 \setminus \Omega_1}(\boldsymbol{X}) &= -d_{\Omega_1}(\boldsymbol{X}) \end{aligned} \tag{2.8}$$

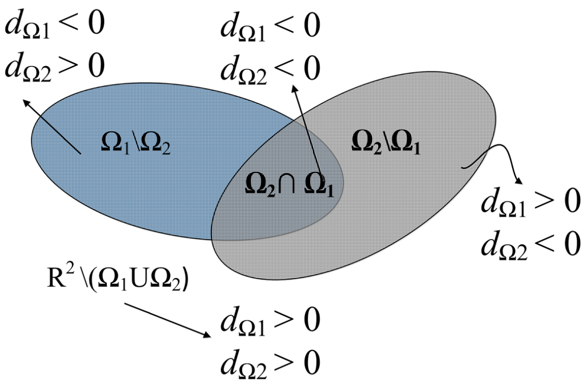
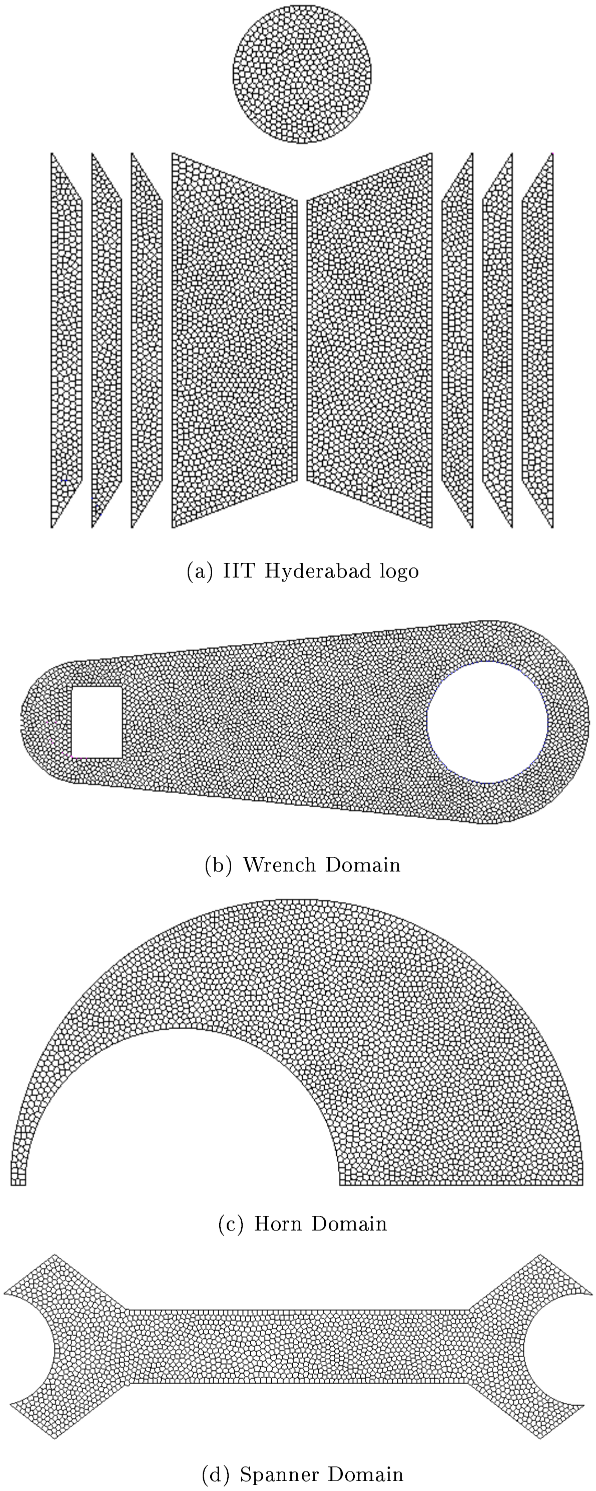


Figure 2.9: Sign property of the distance function for the combined geometry.



(a) IIT Hyderabad logo

(b) Wrench Domain

(c) Horn Domain

(d) Spanner Domain

Figure 2.10: Sample $n-$ sided polygonal meshes generated using CVT procedure.

# Chapter 3

# Conforming interpolants on polygons

Though there are many interpolants defined over polygonal domain, we prefer to use Laplace interpolant in our analysis owing to its ease and simplicity. For a set of $\mathcal{N}$ nodes $\boldsymbol{P}_I = (P_{Ix}, P_{Iy})$ in a domain $\Omega \in \mathbb{R}^2$. At any point $\boldsymbol{P} = (P_x, P_y)$ inside $\Omega$ or on its boundary $\partial\Omega$ a set of associated interpolants $\phi_I(\boldsymbol{P})$ is defined. Using this, an interpolation scheme for a scalar-valued function $f(\boldsymbol{p})$ can be written as

$$f(\boldsymbol{P}) = \sum_{I=1}^{N} \phi_I f_I(\boldsymbol{P}) \tag{3.1}$$

where $f_I = f(\boldsymbol{P}_I)$ are the function values at the $N$ nodes of the polygon. The function $f(\boldsymbol{P})$, satisfies properties such as partition of unity, interpolation and linear completeness inside the polygon and on the boundaries. Various geometric measures like edge length, signed area, and sine or cosine of the angles at each vertex of the polygon are used to construct polygonal interpolants.

The Laplace natural neighbor interpolant is the simplest and most popular Voronoi based interpolation method on polygonal domains [30–32]. The scheme, originally based on the concept of natural neighbors [33], is widely applicable for polygonal domains owing to its ease of implementation, and ability to account for density distribution of nodes in an discretization. Fig.3.1 shows the Voronoi cells for an added point $\boldsymbol{P}$ within a (canonical) polygonal domain with six nodes.

The Laplace interpolation functions [31] are defined using the geometric properties of the Voronoi cell as

$$\phi_I^{\mathrm{L}}(\boldsymbol{P}) = \frac{w_I(\boldsymbol{P})}{\sum_{J=1}^{N} w_J(\boldsymbol{P})} \tag{3.2}$$

$$w_I(\boldsymbol{P}) = \frac{s_I(\boldsymbol{P})}{h_I(\boldsymbol{P})} \tag{3.3}$$

where $s_I(\boldsymbol{P})$ is the length of the associated Voronoi edge and $h_I(\boldsymbol{P}) = \|\boldsymbol{P} - \boldsymbol{P}_I\|$ is the Euclidean distance from node $\boldsymbol{P}$ to node $I$, see Fig.3.1. As only lengths of the Voronoi cells are taken into account, this method belongs to the class of non-Sibsonian interpolants. The Laplace interpolant $\phi_I^{\mathrm{L}}$ on a canonical domain $\Omega_0$, in which all nodes are regularly distributed on a unit circle, is shown in Fig.3.2.

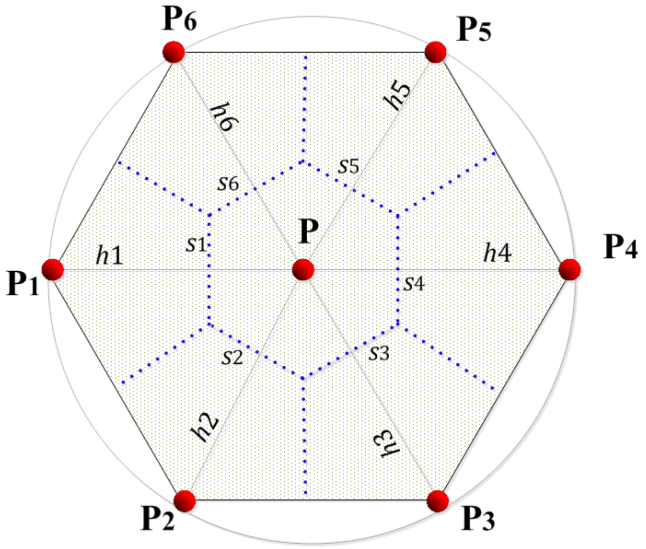


Figure 3.1: Voronoi based geometric measures for the Laplace interpolant: length of the associated Voronoi edge $s_I$, and the Euclidian distance $h_I$ to the evaluation point $\boldsymbol{p}$.

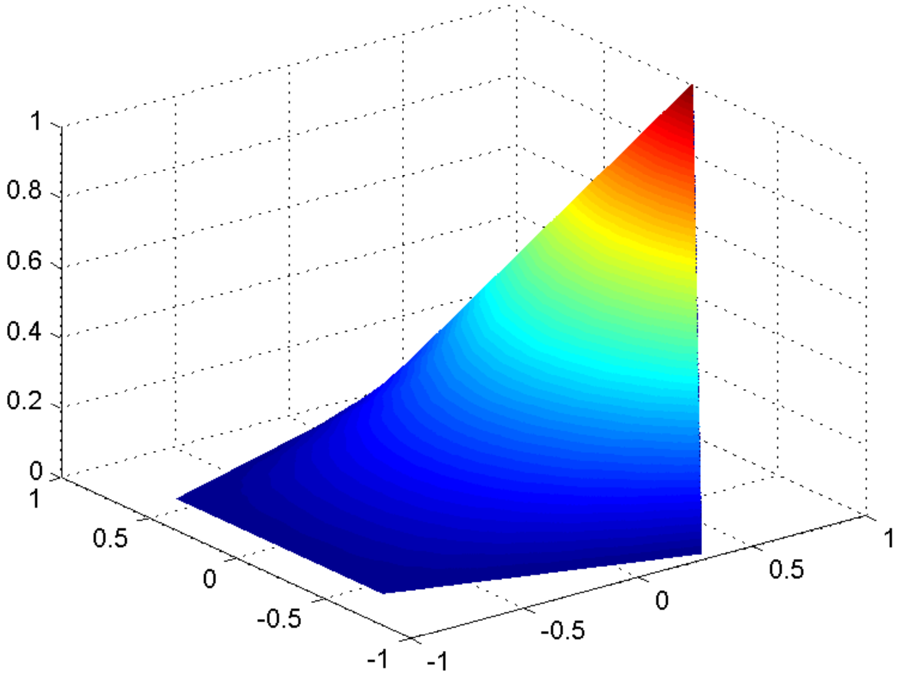


Figure 3.2: Laplace interpolant $\phi_I^{\mathrm{L}}$ on a canonical pentagonal domain .

# Chapter 4

# Governing equations and weak form

We consider the displacement field $u_i(x_j)$ of a body described by an open bounded domain $\Omega \subset \mathbb{R}^2$, see Fig.4.1.

On the Dirichlet boundary $\Gamma_u$ the displacements $\bar{u}_i$ are given, whereas the Neumann boundary $\Gamma_\sigma$ is loaded by the prescribed surface forces $\bar{t}_i$. For small displacements, the governing equations are given by

$$\sigma_{ij,j} + b_i = 0 \quad \text{in} \quad \Omega \tag{4.1a}$$

subjected to boundary conditions

$$u_i = \bar{u}_i \quad \text{on} \quad \Gamma_u \tag{4.1b}$$

$$t_i = \sigma_{ji} n_j = \bar{t}_i \quad \text{on} \quad \Gamma_\sigma \tag{4.1c}$$

with the body force per volume unit $b_i$ and the unit outward normal $n_j$ to $\Gamma_\sigma$.

The linear stress tensor follows from $\sigma_{ij} = \mathbb{D}_{ijkl}\varepsilon_{kl}$ with the strain tensor $\varepsilon_{ij} = \frac{1}{2}[u_{i,j} + u_{j,i}]$ and the material moduli tensor $\mathbb{D}_{ijkl}$. For a homogeneous isotropic material with the Navier-Lamé parameters $\lambda$ and $\mu$, we obtain

$$\mathbb{D}_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu[\delta_{jk}\delta_{jl} + \delta_{il}\delta_{jk}]. \tag{4.2}$$

The weak form is consequently expressed as

$$\int\limits_{\Omega} \sigma_{ij}(u_k)\varepsilon_{ij}(\eta_k)\ d\Omega = \int\limits_{\Omega} b_i\eta_i\ d\Omega + \int\limits_{\Gamma} \bar{t}_i\eta_i\ d\Gamma. \tag{4.3}$$

for the discretization a displacement trial solution of the form $u_i^h \in V = \left[H^1(\Omega)\right]^2$ is chosen together with a set of kinematically admissible test functions $\eta_i^h \in V_0 = \left[H_0^1(\Omega)\right]^2$, vanishing on the Dirichlet boundaries. The discrete form of Eqn.(4.3) leading to the solution for the nodal displacements $\boldsymbol{u}^h$ is given in symbolic notation as

$$\int\limits_{\Omega^h} \boldsymbol{\sigma}\left(\boldsymbol{u}^h\right) : \boldsymbol{\varepsilon}\left(\boldsymbol{\eta}^h\right)\ d\Omega^h = \int\limits_{\Omega^h} \boldsymbol{b}\cdot\boldsymbol{\eta}^h\ d\Omega^h + \int\limits_{\Gamma^h} \bar{\boldsymbol{t}}\cdot\boldsymbol{\eta}^h\ d\Gamma^h. \tag{4.4}$$

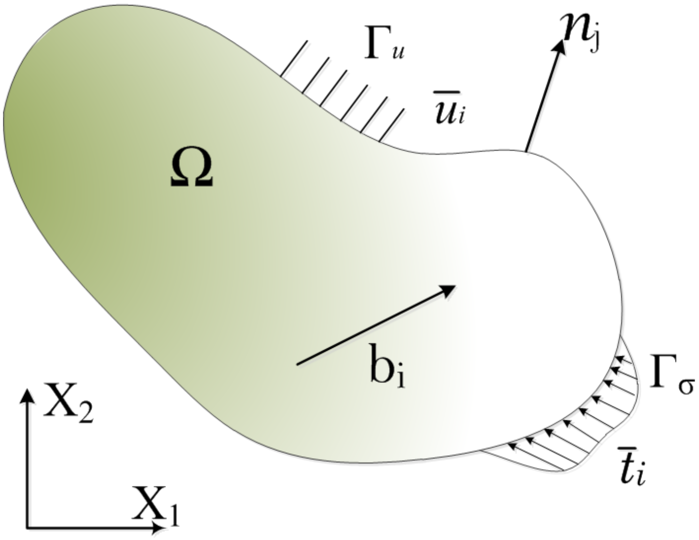


Figure 4.1: Elastostatic boundary value problem.

The trial and test functions are represented as linear combinations of $C^0$-continuous interpolation functions (shape functions),

$$\boldsymbol{u}^h(\boldsymbol{X}) = \sum_{I=1}^{N} \phi_I(\boldsymbol{X}) u_I \quad \text{and} \quad \boldsymbol{\eta}^h(\boldsymbol{X}) = \sum_{I=1}^{N} \phi_I(\boldsymbol{X}) \eta_I, \tag{4.5}$$

On substituting the trial and test functions in Eqn.(4.4) and using the arbitrariness of nodal variations, the discrete system of linear equations

$$K_{IJ} u_J = f_I \tag{4.6}$$

is obtained with the stiffness matrix

$$K_{IJ} = \int\limits_{\Omega^h} B_I^T \hat{\mathbb{D}} B_J \; d\Omega^h \tag{4.7}$$

and the external load vector

$$f_I = \int\limits_{\Gamma^h} \phi_I \bar{t} \; d\Gamma^h + \int\limits_{\Omega^h} \phi_I b \; d\Omega^h \tag{4.8}$$

where the matrix of the shape function derivatives is given by

$$B_I = \begin{bmatrix} \phi_{I,x} & 0 \\ 0 & \phi_{I,y} \\ \phi_{I,y} & \phi_{I,x} \end{bmatrix}$$

and the material moduli tensor in Voigt notation for an isotropic linear material and plane stress condition is given by

$$\mathbb{D} = \frac{E}{1-\nu^2}\begin{bmatrix} 1 & \nu & 0 \\ \nu & 1 & 0 \\ 0 & 0 & \frac{1-\nu}{2} \end{bmatrix}$$

# Chapter 5

# Numerical integration of the weak form

Numerical integration of the Galerkin weak form is required to be performed over the polygonal domain for evaluating the integrals given in Eqn.(4.7) and Eqn.(4.8). The standard Gaussian integration rule is used for finite elements and for mesh free methods based on background cells. However for non-polynomial type coordinates the Gaussian quadrature cannot guarantee exact results. Presently the state of the art includes the following methods for performing numerical integration over polygonal domains:

- Partitioning of the physical polygonal element domain $\Omega$ into $N$ triangular subdomains and performing numerical quadrature on the physical subdomains [34, 35],
- Partitioning of the canonical polygonal element domain $\Omega_0$ into $N$ triangular subdomains and performing numerical quadrature on the canonical subdomains with isoparametric mapping [35, 36],
- Cubature rules for irregular $N$-gons [37, 38] based on triangles [39, 40] or conformal mapping [11, 12],
- Generalized quadratures rules [41] on triangles or polygons based on symmetry groups and numerical optimization [14, 42–44].

In the present work we use the first approach, we point out that the partition of $n$-gon into $N$ sub-triangles is used solely for the purpose of numerical integration, and hence within any sub-triangles, both the shape function $\phi_I$ as well as its gradient $\phi_{(I,J)}$ $(I = 1, 2, ...., N,\ \ J = 1, 2)$ are non-zero. We use centroid of element to partition it into $N$ triangles for the purpose of numerical integration. Thus, the numerical integration of a scalar function $\Phi$ over $\Omega$ (a $n$-gon) is written as

$$\int_{\Omega} \Phi \; d\Omega = \sum_{J=1}^{N} \int_{\Omega_J} \Phi \; d\Omega \tag{5.1}$$

$$= \sum_{J=1}^{N} \int_{0}^{1} \int_{0}^{1-\xi} \Phi \; |\boldsymbol{J}_J| \; d\xi \; d\eta \tag{5.2}$$

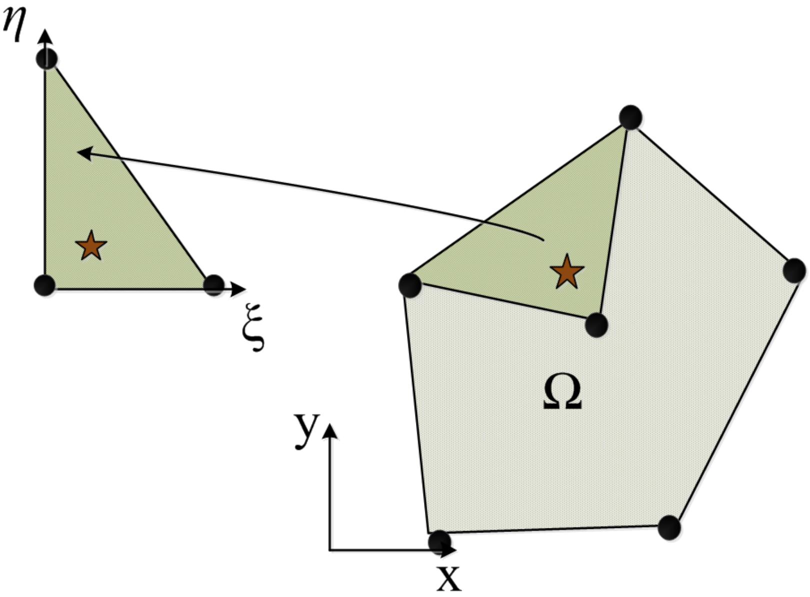


Figure 5.1: Numerical integration based on partition of the physical element domain $\Omega$ and mapping of quadrature points to a parameterized triangular domain $\bar{\Omega}_0$.

where, $N$ is number of sub partition (No. of triangles), $\Omega_J$ is domain and $\boldsymbol{J}_J$ is jacobian for $J^{th}$ sub partition.

# Chapter 6

# Error estimation and Adaptivity

Although finite element analysis is most widely used tool for the solution of the large class of engineering problems characterized by the partial differential equations, the accuracy of the solution is always a concern. From a global point of view the inaccuracy may be attributed to the modeling drawbacks of the finite element analysis since it is practically impossible to characterize infinite degrees of freedom system by a discrete finite element model. This modeling deficiency usually results in a lower bound of solution which is manifested by stiffening in structural mechanics problems. It is important to devise methodologies which can quantify the measure of reliability of solution, i.e. which can measure the amount by which the modeled solution actually drifts from a bench mark solution of the same problem. Adaptive finite element techniques seek to construct these reference solutions, define error norms and in general create a more accurate and reliable numerical solution by using a feed back strategy incorporating these reference solution and error norms. The computed error could be based on a *priori* or a *posteriori* estimators. The later has gained more popularity because of its robustness. The two types of estimate serve different purposes. The main feature of a *priori* estimates is that they tell us the order of convergence of a given finite element method, that is, they tell us that the finite element error $\|\boldsymbol{u} - \boldsymbol{u}^h\|$ in some norm $\|\cdot\|$ is $O(h^{(\alpha)})$, where h is the (maximum) mesh size and $\alpha$ is a positive integer. The goal of these estimates is to give us a reasonable measure of the efficiency of a given method by telling us how fast the error decreases as we decrease the mesh size. In contrast, a posteriori estimates use the computed solution $\boldsymbol{u}^h$ in order to give us an estimate of the form $\|\boldsymbol{u} - \boldsymbol{u}^h\| \leq tol$ where $tol$ is simply a number. These estimates accomplish two main goals. First, they are able to give us a much better idea of the actual error in a given finite element computation than are a priori estimates. Secondly, they can be used to perform adaptive mesh refinement. In adaptive mesh refinement, a posteriori error estimators are used to indicate where the error is particularly high, and more mesh intervals are then placed in those locations. A new finite element solution is computed, and the process is repeated until a satisfactory error tolerance is reached.

### 6.0.1 *a-posteriori* error estimate

To assess the ability of polygonal finite elements to represent linear displacement fields, we first perform displacement patch test. We use $L^2(\Omega)$ and $H^1(\Omega)$ norms of the displacement error to assess the performance of the polygonal finite element method. The $L^2(\Omega)$ and $H^1(\Omega)$ error norms are defined as follows

$$\|\boldsymbol{u} - \boldsymbol{u}^h\|_{L^2(\Omega)} = \sqrt{\int_\Omega [u_i - u_i^h]^T \ [u_i - u_i^h] \ d\Omega} \tag{6.1}$$

$$\|\boldsymbol{u} - \boldsymbol{u}^h\|_{H^1(\Omega)} = \|\boldsymbol{u} - \boldsymbol{u}^h\|_{L^2(\Omega)} + \sqrt{\int_\Omega [u_{i,j} - u_{i,j}^h]^T \ [u_{i,j} - u_{i,j}^h] \ d\Omega} \tag{6.2}$$

where $\boldsymbol{u}$ and $\boldsymbol{u}^h$ are the exact and numerical displacements respectively.

In finite element methods, usually $C^o$ (or $C^1$ for some plate and shell elements) functions are used for the interpolation of the unknown function (i.e. displacements); this however entails a discontinuity in the derivatives across the element boundaries which give rise to discretization errors. A post-processing procedures could be devised which could use these derivatives discontinuities to create smoothed functions which could be used in place of the exact solution in the error estimation process. Many such post processing procedures have been reported in the literature so far i.e Hinton and Campbell [45], Oden and Brauchli [46] and Zienkiwicz and Zhu [19]. They may be known collectively as *smoothing operations.* An important concept is that the discrepancy between element by element field and smoothed field can serve as a measure of discretization error. For this purpose, and then for deciding how the discretization can be improved, the smoothed field is regarded as the most accurate result the current mesh can provide. The more accurate alternate field can be obtained by various techniques like nodal averaging, global smoothing or patch recovery. In present work we use *patch recovery* technique to obtain smoothed quantities.

Patch recovery technique was proposed by *Zienkiewicz* and *Zhu* (1992a, 1992b). The basis of the method lies in the least square fit of an unknown polynomial over a patch of elements at sampling points. Either stress or strain can be smoothed. A small number of contiguous elements, called a *patch*, is selected. Fig.6.1 shows a polygonal mesh containing 10 elements and 21 nodes. There are several patches associated with this mesh. For example consider node 6, a patch associated with node 6 is formed by elements $1, 5$ *and* 4 similarly patch associated with node 7 is formed by elements $1, 2$ *and* 5. Fig.6.2 shows the patches associated with node 6 and 7.
Consider a patch associated with node 6, see Fig6.2(a), choosing stress to be smoothened, we represent smoothed stress field, $\boldsymbol{\sigma}^\bigstar$ over the patch as,

$$\boldsymbol{\sigma}^\bigstar = [P]\{a\} \tag{6.3}$$

Where $[P]$ is the basis of assumed polynomial, and $\{a\}$ contains generalized co-ordinates to be determined. Different values of the $a_i$ in $\{a\}$ are obtained for different stress components. To determine $\{a\}$ and thus define smooth stress field, we select a patch as in Fig 6.2(a) and sample element stress field $\boldsymbol{\sigma}$ at locations where it is likely to be most accurate. These points are generally gauss points of

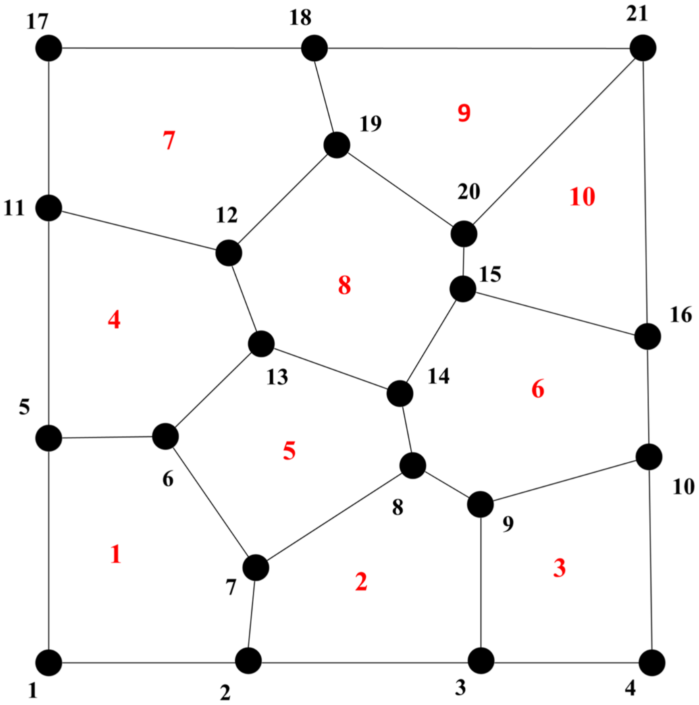


Figure 6.1: Polygonal mesh of 10 elements and 21 nodes.

various integration rules, shown in Fig.6.3 . We then make a least square fit of expansion Eqn.(6.3) to finite element solutions $\boldsymbol{\sigma}$ obtained at sampling points over the patch. Thus we minimize the functional

$$\Pi = \sum_{j=1}^{nsp} (\boldsymbol{\sigma} - [P]\{a\})^2 \tag{6.4}$$

This implies

$$[A]\{a\} = \{b\} \tag{6.5}$$

*where*

$$[A] = \sum_{j=1}^{nsp} [P]^T \ [P] \tag{6.6}$$

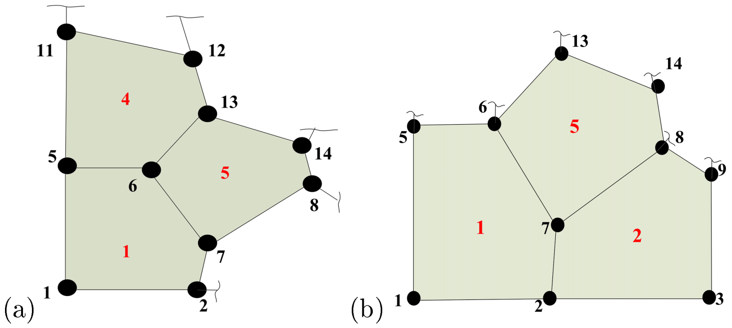


Figure 6.2: (a)Patch associated with node 6 of a mesh,(b) Patch associated with node 7 of a mesh.

$$\{b\} = \sum_{j=1}^{nsp} \; [P]^T \;\; \boldsymbol{\sigma} \tag{6.7}$$

*where,*
*nsp* is number of gauss points in a patch.
When $\{a\}$ has been computed, the smoothed stress $\boldsymbol{\sigma}^{\bigstar}$ can be evaluated at any point inside the patch by substituting $x$ and $y$ coordinates of the point in Eqn.(6.3).
A element $i$ can appear in more than one patch, i.e, consider a mesh in fig 6.1, element 1 is part of patch formed by node 6 as well as part of patch formed by node 7 (See Fig.6.2), so it is important to keep track of number of times a particular element $i$ has appeared in various patches. The smoothed stress field $\boldsymbol{\sigma}^{\bigstar}$ in elements have to be averaged before using them in error estimation.
A finite element solution contains enough information to estimate its own error. That is, a *posteriori* error estimate is possible. These estimates use the computed solution $\boldsymbol{u}^h$ in order to give us an estimate of the form $\|\boldsymbol{u} - \boldsymbol{u}^h\| \leq tol$ which can be used to perform adaptive mesh refinement. We in our analysis use a posteriori averaging based error indicator using explicitly calculated smoothened stresses,$\boldsymbol{\sigma}^{\bigstar}$ presented by *Zienkiewicz* and *Zhu.* The estimate is based on element by element field and an smooth field obtained as in Eqn.(6.3). The *Zienkiewicz* and *Zhu* often called **ZZ** or $\mathbf{Z^2}$ error estimate can be defined as follows,
The *global strain energy norm* $\|U\|$ is defined as

$$\|U\| = \left[ \sum_{i=1}^{m} \int \{\boldsymbol{\sigma}\}_i^T \;\; \mathbb{D}^{-1} \;\; \{\boldsymbol{\sigma}\}_i \;\; d\Omega \right]^{\frac{1}{2}} \tag{6.8}$$

$$\|e\| = \left[ \sum_{i=1}^{m} \int (\{\boldsymbol{\sigma}\}_i^{\bigstar} - \{\boldsymbol{\sigma}\}_i)^T \;\; \mathbb{D}^{-1} \;\; (\{\boldsymbol{\sigma}\}_i^{\bigstar} - \{\boldsymbol{\sigma}\}_i) \;\; d\Omega \right]^{\frac{1}{2}} \tag{6.9}$$

*where* m is number of elements in the region of a structure whose error is to be estimated.
As an alternative to $\|u\|$ and $\|e\|$ one can work with $L_2$ norm quantities. They are obtained from the foregoing expression by omitting the weighting matrix $\mathbb{D}$.

$$\|U\| = \left[ \sum_{i=1}^{m} \int \{\boldsymbol{\sigma}\}_i^T \;\; \{\boldsymbol{\sigma}\}_i \;\; d\Omega \right]^{\frac{1}{2}} \tag{6.10}$$

$$\|e\| = \left[ \sum_{i=1}^{m} \int (\{\boldsymbol{\sigma}\}_i^{\bigstar} - \{\boldsymbol{\sigma}\}_i)^T \;\; (\{\boldsymbol{\sigma}\}_i^{\bigstar} - \{\boldsymbol{\sigma}\}_i) \;\; d\Omega \right]^{\frac{1}{2}} \tag{6.11}$$

*where,* $\{\boldsymbol{\sigma}^{\bigstar}\}_i$ is smooth stress field obtained from *patch recovery* technique and
$\{\boldsymbol{\sigma}\}_i$ is finite element stress.
Relative error $\eta$ used to quantify the discretization error over a patch of elements or the entire mesh, can be defined as

$$\eta = \left[ \frac{\|e\|^2}{\|U\|^2 + \|e\|^2} \right]^{\frac{1}{2}} \tag{6.12}$$

The possible range of $\eta$ is $0 < \eta < 1$.

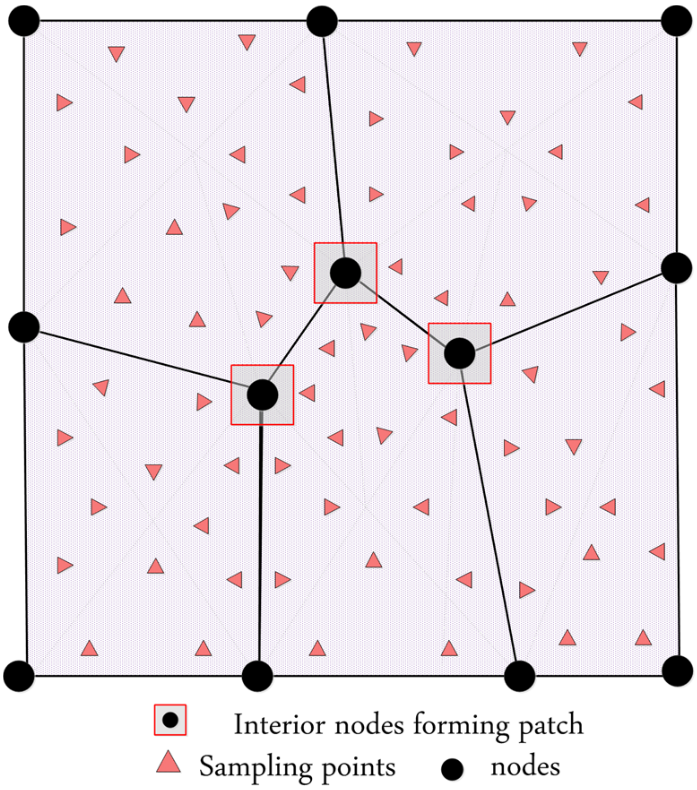


Figure 6.3: Figure showing location of sampling points and nodes in a simple mesh of polygons

### 6.0.2 Adaptive Meshing

The goal of adaptive meshing is to achieve a desired accuracy by revising a mesh, where necessary and to extent necessary. It is a iterative process. Automation of the process requires numerical indication on where and how to revise the discretization and a termination criteria. In our analysis we have used *relative error norm*, $\eta$ described in Section 6.0.1, as an criteria for adaptivity.

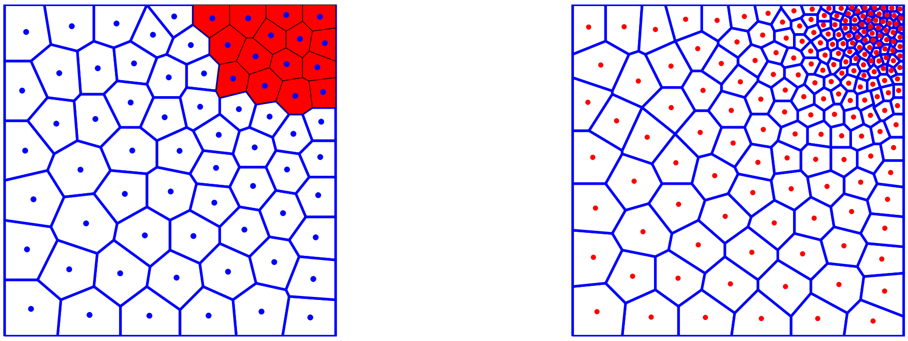
A possible termination criteria is that final value of $\eta$ must not exceed an allowable value $\eta_{all}$ in the final mesh. The average value of $\|U\|^2 + \|e\|^2$ per element, in combination with $\eta_{all}$, provides an allowable value of error, $\|e_i\|_{all}$ in an element. Thus from Eqn.(6.12), with $m$ the number of elements in the mesh, the allowable error in a single element can be calculated as

$$\|e_i\|_{all} = \eta_{all} \left[ \frac{\|U\|^2 + \|e\|^2}{m} \right]^{\frac{1}{2}} \tag{6.13}$$

We then calculate $\xi$ for each element, which is ratio of actual value of $\|e_i\|$ in a typical element to allowable value is

$$\xi_i = \frac{\|e_i\|}{\|e_i\|_{all}} \tag{6.14}$$

The result $\xi_i > 1$ indicates that more elements are needed in that location. After first cycle of analysis if any element has $\xi_i < 1$, is ignored with attention directed towards refinement of elements for which $\xi_i > 1$. With $\chi_i$ a characteristic element dimension, the desired new size of the element $i$ is taken as



(a) Initial mesh with point set ● $P_o$, elements marked red have $\xi_i > 1$

(b) Refined mesh with updated point set ● $P_{new}$

Figure 6.4: Adaptive refinement of a block.

$$(\chi_i)_{new} = \frac{(\chi_i)_{old}}{\xi_i} \tag{6.15}$$

In our work, area of the element is considered as the characteristic element dimension, $\chi$. Consider initial coarse mesh as in fig 6.4(a) formed by point set $P_0$. Compute finite element solution $\boldsymbol{\sigma}$ for this mesh and calculate allowable error $\|e_i\|_{all}$, (Eqn.(6.13)) for each element in the mesh. Further calculate $\xi_i$ using Eqn.(6.14), for all the elements in the mesh. Fig.6.4(a) highlights the elements of initial mesh with $\xi > 1$, indicating these elements needs to be refined. For these elements we calculate $(\chi_i)_{new}$, as in Eqn.(6.15). To reduce element size to $(\chi_i)_{new}$, extra seeds (say $P_i$) are randomly introduced in the erroneous elements i.e., $\xi_i > 1$. The initial point set $P_0$ is updated with newly introduced seeds $P_i$ to get a updated point set, $P_{new}$. The new mesh is generated with this updated point set is shown in fig6.4(b). This process is repeated until $\eta < \eta_{all}$ .

The steps in the iterative adaptive strategy is explained in algorithm 2.

**Algorithm 2** Algorithm for Adaptive strategy

```
1:  Create a initial coarse mesh in $\Omega$ with random initial point set $P_o$
2:  Compute the finite element solution $\boldsymbol{\sigma}$.
3:  Compute *relative error* $\eta$ for the mesh as in Eqn.(6.12).
4:  **if** $\eta > \eta_{all}$ **then**
5:      **for all** Elements **do**
6:          Compute *allowable error* $\|e_i\|_{all}$ as in Eqn.(6.13).
7:          Compute $\xi_i$ as in Eqn(6.14)
8:          **if** $\xi_i \ > \ 1$ **then**
9:              Reduce the element size to $(\chi_i)_{new}$ as in Eqn.(6.15) by introducing new set of random
    points $P_i$ in that element
10:             Update $P_{new} \leftarrow P_o + P_i$
11:         **end if**
12:     **end for**
13: **end if**
14: Generate new mesh with updated point set $P_{new}$
15: Go to step 2
```

# Chapter 7

# Numerical Example

## 7.1 Numerical Examples

In this section, first we study the ability of proposed method to represent linear displacement field via displacement patch test. The adaptive procedure given above is employed to refine initial mesh, in order to get desired mesh ($\eta < \eta_{all}$). We consider three numerical examples under plane stress conditions to demonstrate the proposed method. Numerical integration on an $n$ sided polygon is performed using $3X3$ Gauss quadrature rule on $n$ sub - triangles of a $n$ sided polygon (See Fig.17). In all the considered examples we use modulus of elasticity $E = 70000Mpa$ and Poisson's ratio, $\mu = 0.3$.

### 7.1.1 Displacement Patch test

As a first example, we study the ability of polygonal finite elements to represent linear displacement fields. We consider a unit square domain, as shown in Fig.19. The test is performed by applying displacements of $u_x = x_i$ and $u_y = y_i$ on the boundary nodes of a unit square domain. The displacement error norms, $L_2$ and $H_1$ are calculated for four different mesh shown in Fig.7.2. The values of $L_2$ and $H_1$ noms are tabulated in table 7.1. Results reveal that patch test is passed. The $L_2$-norm and $H_1$-norm of displacement error is shown in Table 7.1.

Table 7.1: $L_2$ and $H_1$ norm for displacement patch test

| Mesh | $dof$ | $L_2$ norm | $H_1$ norm |
|---|---|---|---|
| a | 42 | 2.21E-05 | 2.84E-03 |
| b | 82 | 1.62E-05 | 5.49E-03 |
| c | 160 | 1.53E-05 | 6.44E-03 |
| d | 318 | 9.74E-06 | 7.03E-03 |

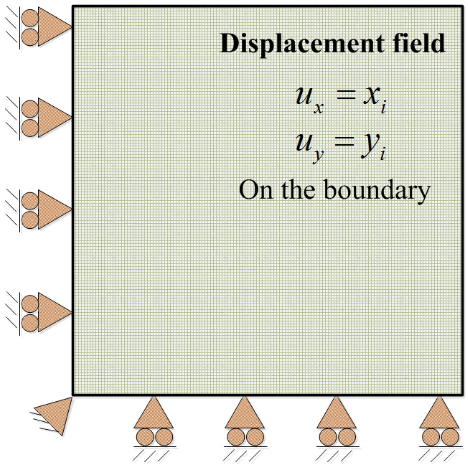


Figure 7.1: Unit square domain subjected to displacement field of $u_x = x_i$ and $u_y = y_i$ on the boundary.

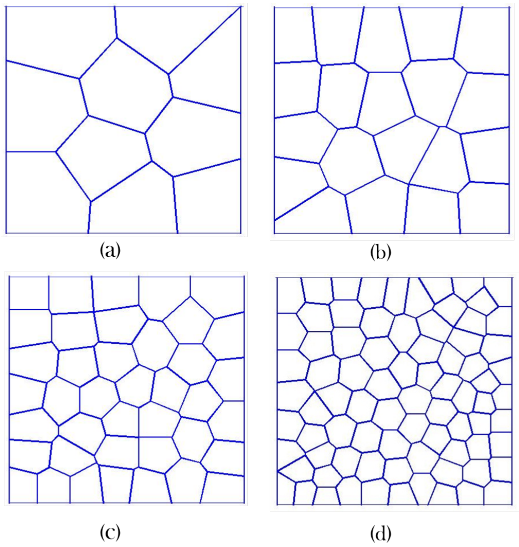


Figure 7.2: Different mesh considered for displacement patch test:(a)Mesh with 10 elements,(b) Mesh with 20 elements,(c)Mesh with 40 elements,(d) Mesh with 80 elements.

### 7.1.2 Plate with a Circular hole

A non - convex plate with a traction - free circular hole is considered (edge length $2\ell$, hole radius $a = \frac{\ell}{6}$). Fig.7.3 shows the quarter plate loaded by unidirectional tension $\sigma_0 = 100 Mpa$ in $x$ - direction on the right edge. Due to symmetry, Dirchlet boundary conditions are imposed along $AB(u_y = 0)$ and $DE(u_x = 0)$. In polar coordinates the exact stress field for the infinite plate is given by

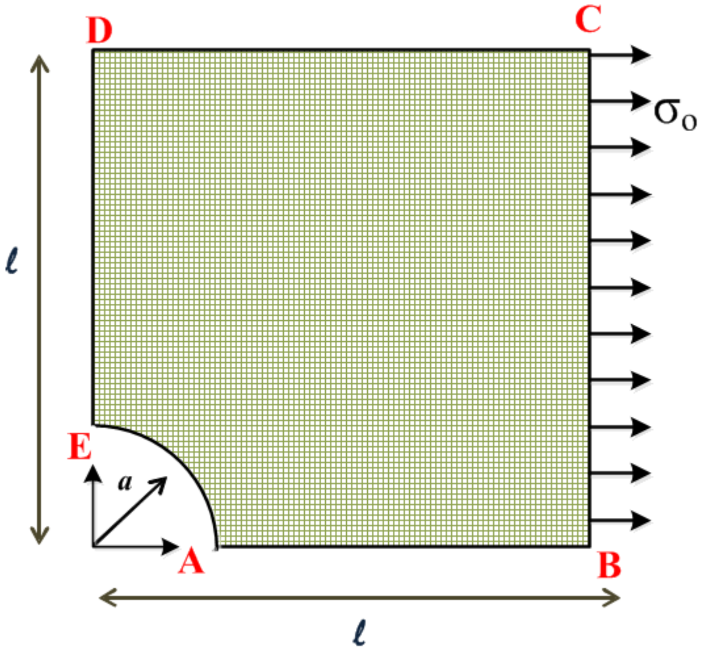


Figure 7.3: Quarter plate with circular hole under tension with symmetry axes

$$\begin{aligned}
\frac{\sigma_{xx}(r,\theta)}{\sigma_0} &= 1 - \frac{a^2}{r^2}\left[\frac{3}{2}\cos 2\theta + \cos 4\theta\right] + \frac{3a^4}{2r^4}\cos 4\theta \\
\frac{\sigma_{yy}(r,\theta)}{\sigma_0} &= -\frac{a^2}{r^2}\left[\frac{1}{2}\cos 2\theta - \cos 4\theta\right] - \frac{3a^4}{2r^4}\cos 4\theta \\
\frac{\sigma_{xy}(r,\theta)}{\sigma_0} &= -\frac{a^2}{r^2}\left[\frac{1}{2}\sin 2\theta + \sin 4\theta\right] + \frac{3a^4}{2r^4}\sin 4\theta
\end{aligned} \tag{7.1}$$

The corresponding displacement components are

$$\begin{aligned}
u_x(r,\theta) &= \frac{a}{8\mu}\left[\frac{r}{a}(\kappa+1)\cos\theta + 2\frac{a}{r}((1+\kappa)\cos\theta \right. \\
&\qquad \left. + \cos 3\theta) - 2\frac{a^3}{r^3}\cos 3\theta\right]
\end{aligned} \tag{7.2}$$

$$\begin{aligned}
u_y(r,\theta) &= \frac{a}{8\mu}\left[\frac{r}{a}(\kappa-3)\sin\theta + 2\frac{a}{r}((1-\kappa)\sin\theta \right. \\
&\qquad \left. + \sin 3\theta) - 2\frac{a^3}{r^3}\sin 3\theta\right]
\end{aligned} \tag{7.3}$$

with $\kappa = 3 - 4\nu$. In the numerical computations, we consider $a = 10mm$, $\ell = 60mm$ and the Laplace interpolant $\phi_I^{\mathrm{L}}$. As a termination criteria we have chosen $\eta_{all} = 5\%$. Initial discretization being the

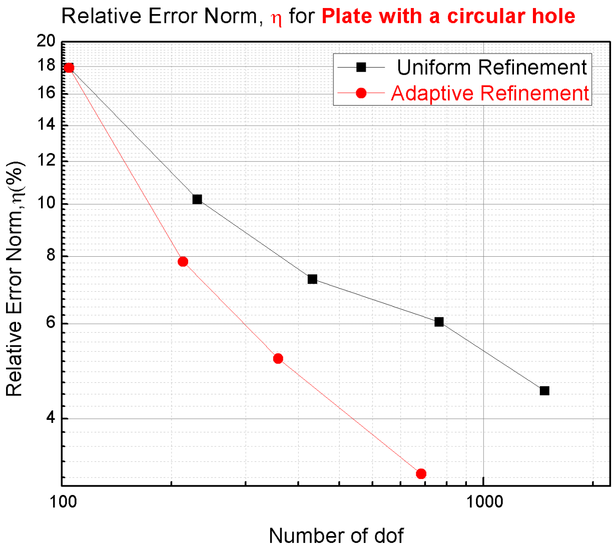
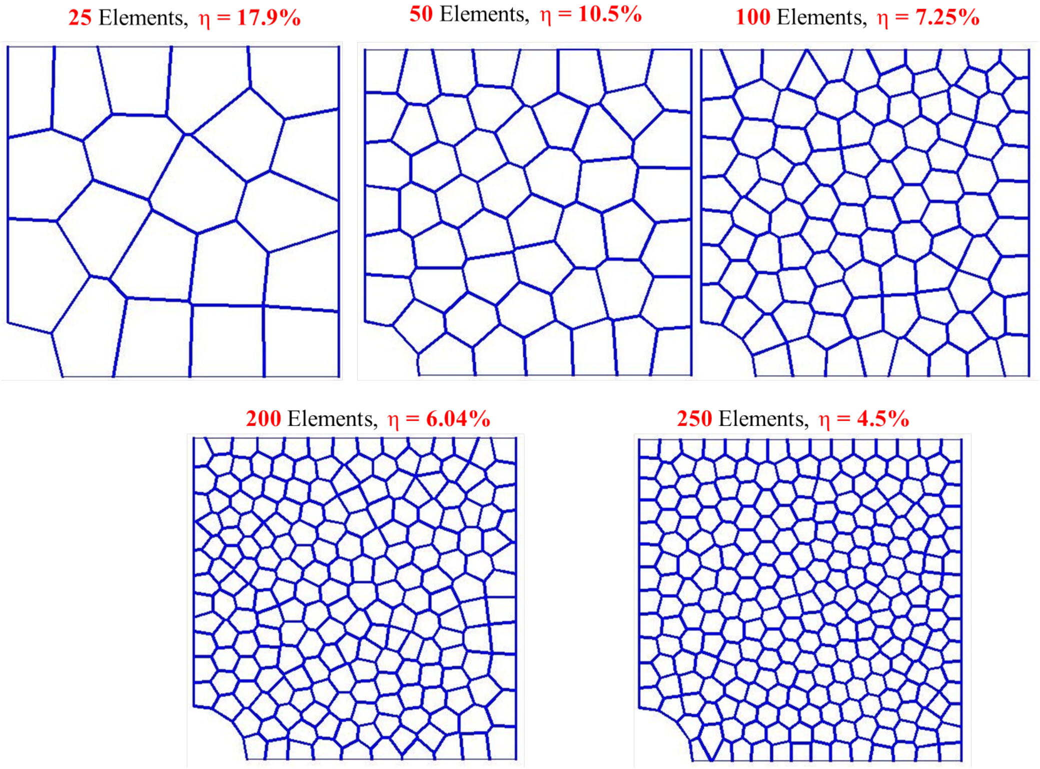


Figure 7.4: Plot of relative error norm $\eta$ for Plate with a circular hole problem.

coarsest has 25 elements with 104 $dof$. We perform uniform and adaptive refeinement on this initial coarse mesh. Fig.7.5 and Fig.7.6 shows the different meshes obtained due to uniform and adaptive refinements respectively.

Fig.7.1.2 shows a plot of relative error norm *vs* number of dof for a plate with a hole example. It is seen in graph that initial coarse mesh having 104 dof, has a $\eta = 17.9\%$. Adaptive refinement reaches the criteria of $\eta < \eta_{all}$(5%in this case) at 712 dof, while it takes 1400 dof for uniform refinement to satisfy the same criteria. We see that adaptive refinement converges at relatively higher order than uniform refinement. The stress plots for the Plate with a hole example are shown in Fig.7.7.



25 Elements, $\eta = 17.9\%$

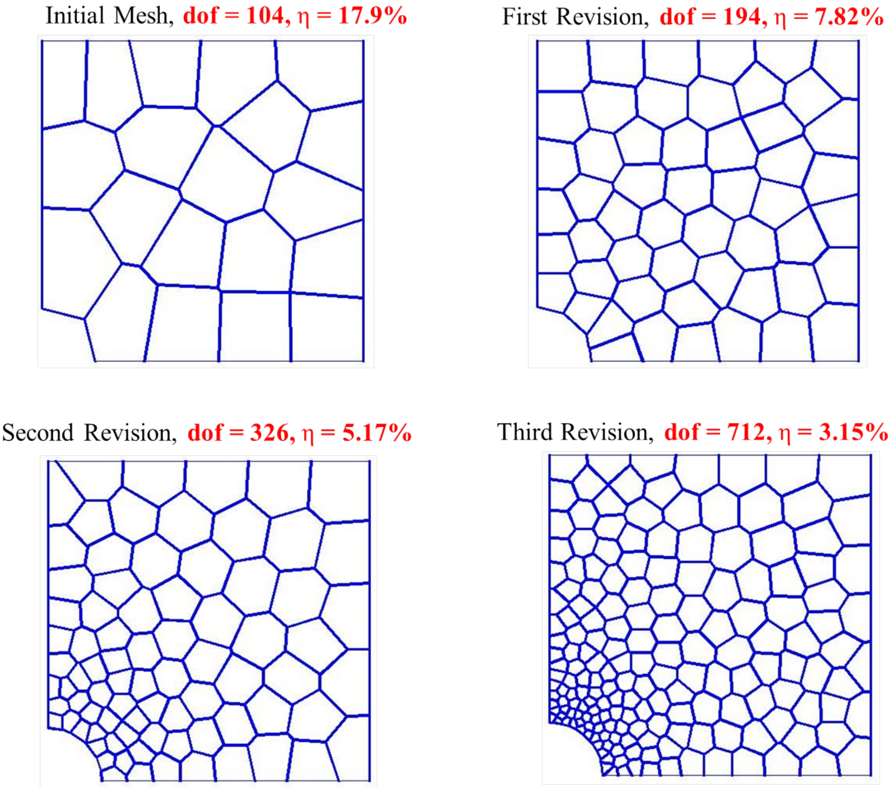
50 Elements, $\eta = 10.5\%$

100 Elements, $\eta = 7.25\%$

200 Elements, $\eta = 6.04\%$

250 Elements, $\eta = 4.5\%$

Figure 7.5: Meshes generated due to uniform refinement of the plate with a circular hole domain



Initial Mesh, dof = 104, $\eta = 17.9\%$

First Revision, dof = 194, $\eta = 7.82\%$

Second Revision, dof = 326, $\eta = 5.17\%$

Third Revision, dof = 712, $\eta = 3.15\%$

Figure 7.6: Meshes generated due to adaptive refinement of the plate with a circular hole domain

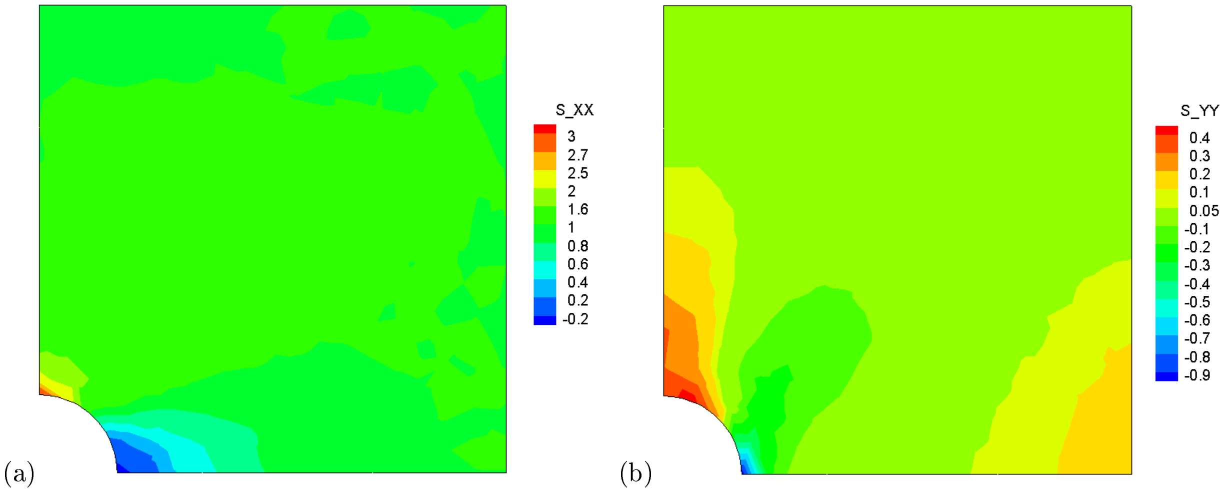


Figure 7.7: Stress plot for plate with a circular hole problem.

### 7.1.3 L shaped domain

A $L$-shaped specimen of the dimensions as shown in Fig.7.8 is considered. The specimen is subjected to a prescribed displacement($u_x = 10mm$) and ($u_y = 10mm$) along the two legs. To prevent rigid body motion all the deformation degrees of freedom at the re-entrant corner $A$, are additionally fixed. In the numerical computations, we consider, $\ell = 100mm$ with the Laplace interpolant $\phi_I^{\mathrm{L}}$. As a termination criteria we have chosen $\eta_{all} = 10\%$. Initial discretization being the coarsest has 25 elements in the domain with 108 $dof$. We perform uniform and adaptive refinement on this coarse mesh. Fig.7.10 and Fig.7.11 shows the different meshes obtained due to uniform and adaptive refinement. Fig.7.9 shows a plot of relative error norm $vs$ number of dof for a L example. It is

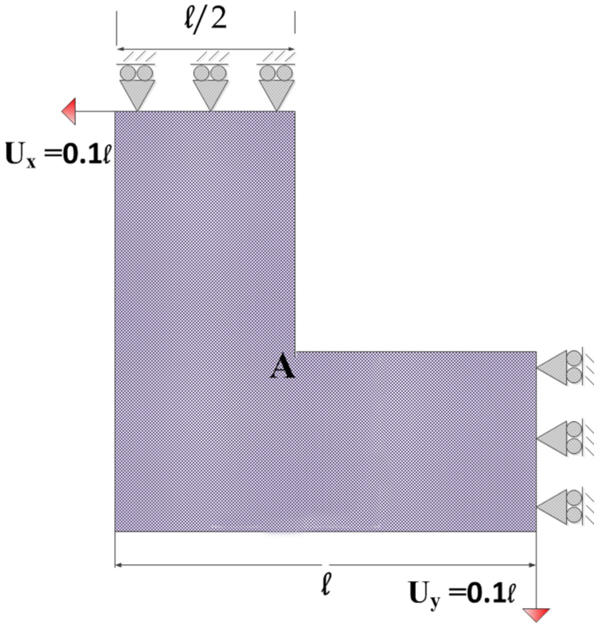


Figure 7.8: $L$ -shaped specimen.

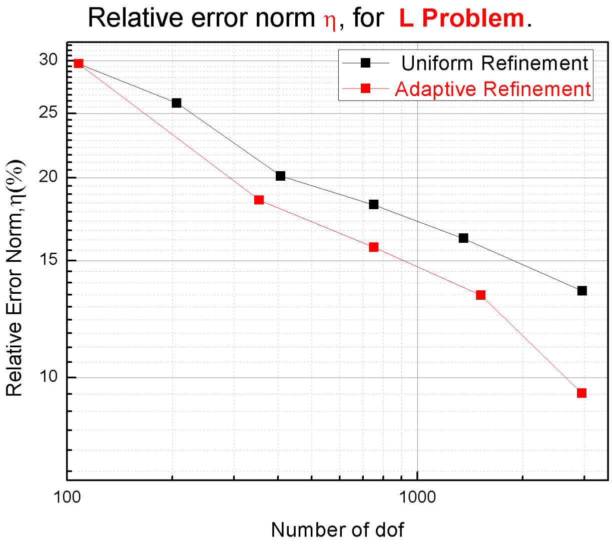
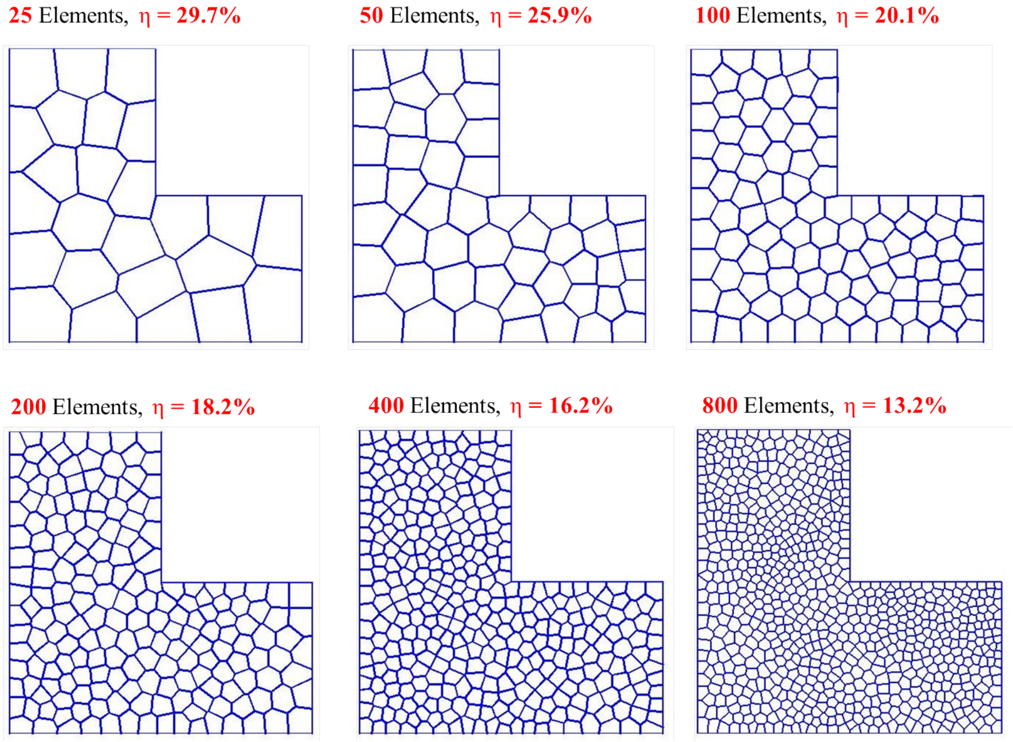


Figure 7.9: Plot of relative error norm $\eta$ for L problem.

seen from the graph that initial coarse mesh having 108 dof, has error of 29.7%. We also note that there is no much difference between convergence rates of both the methods during initial stages. This can be attributed to the fact that, initially most of the elements in the L domain are erroneous, i.e, $\xi_i > 1$, so that even in adaptive refinement most of the elements are refined. After certain refinements adaptive refinement steeps off and reaches the criteria of $\eta < \eta_{all}$(10%in this case) at 2950 dof, while even at 3000 dof for uniformly refined mesh possesses error of 13.2%, which is greater than $\eta_{all}$.

The stress plots for the L domain example are shown in Fig.7.12.



25 Elements, η = 29.7%

50 Elements, η = 25.9%

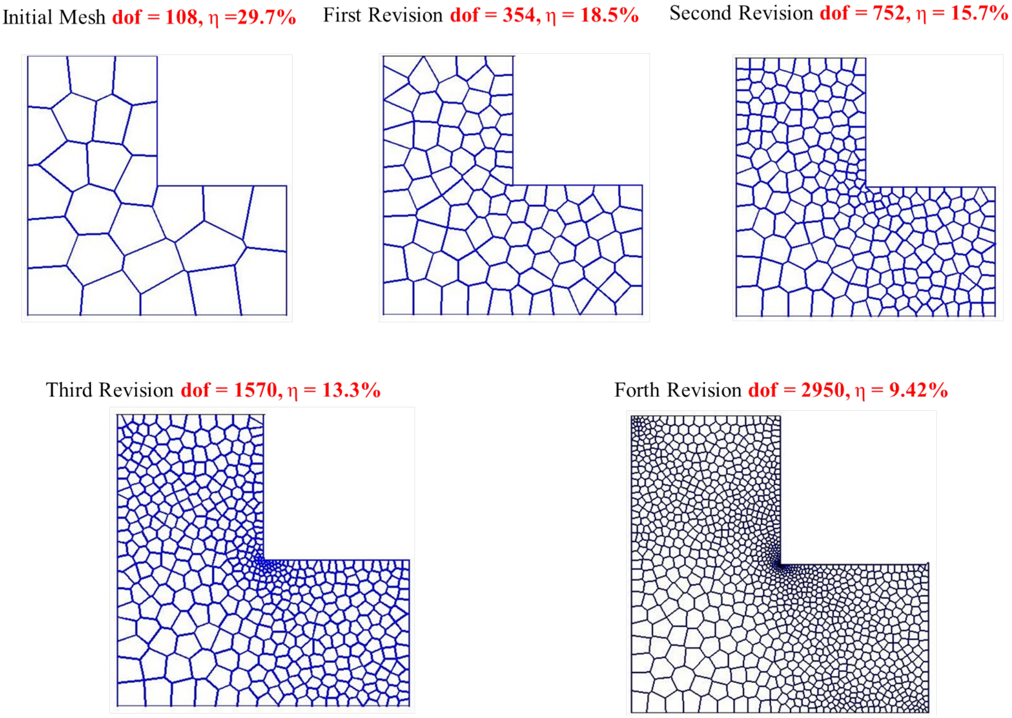
100 Elements, η = 20.1%

200 Elements, η = 18.2%

400 Elements, η = 16.2%

800 Elements, η = 13.2%

Figure 7.10: Meshes generated due to uniform refinement of $L$ shaped domain.



Initial Mesh dof = 108, η =29.7%

First Revision dof = 354, η = 18.5%

Second Revision dof = 752, η = 15.7%

Third Revision dof = 1570, η = 13.3%

Forth Revision dof = 2950, η = 9.42%

Figure 7.11: Meshes generated due to adaptive refinement of $L$ shaped domain

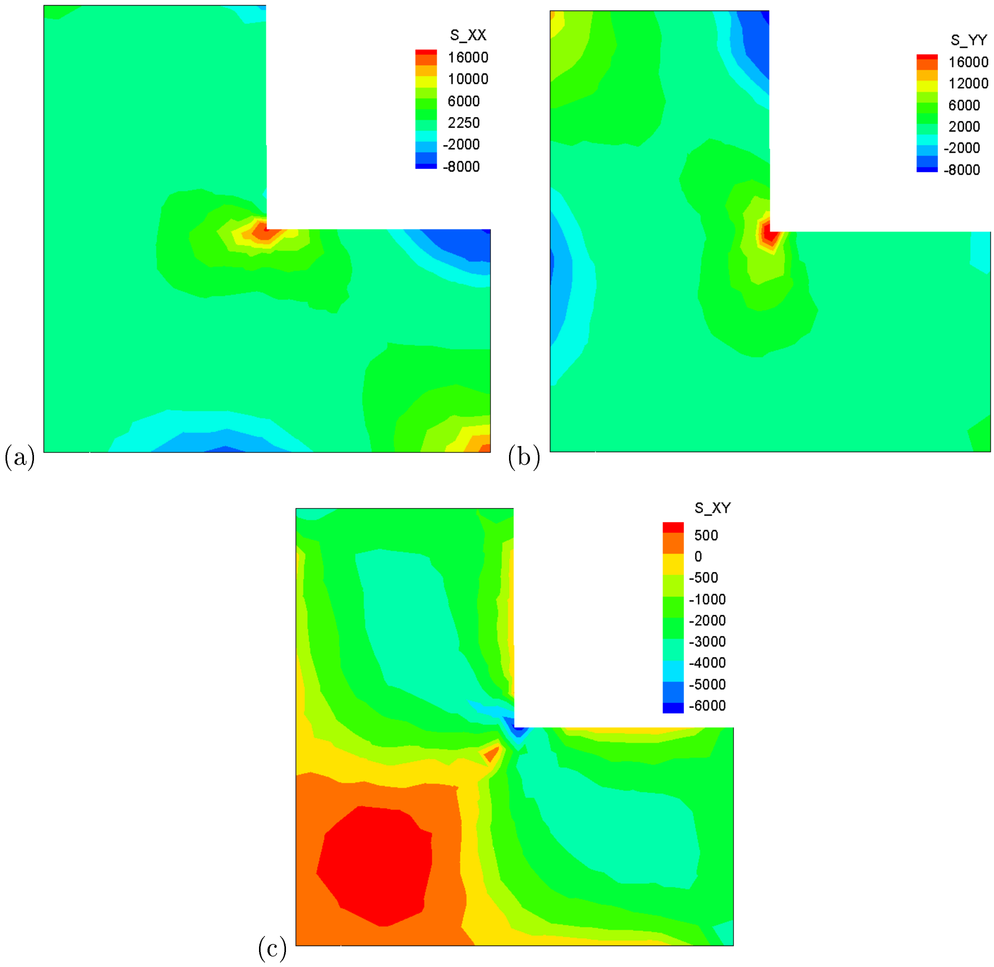


Figure 7.12: Stress plot for L Problem.

### 7.1.4 Bracket problem

As a last example we consider a mechanical connection as shown in Fig7.13.Bracket has 2 bolt holes and all degrees of freedom along the bolt hole is restrained and is subjected to 2 point loads at the tips. In the numerical computations, we consider $\ell = 100mm$, $a = 50mm$ and bolt holes of dia $25mm$,i.e, $b = 12.5mm$, $P = 500N$ with the Laplace interpolant $\phi_I^{\mathrm{L}}$. As a termination criteria we have chosen $\eta_{all} = 15\%$. Initial discretization being the coarsest has 25 elements with 108 $dof$. We perform uniform and adaptive refinement on this mesh. Fig.7.15 and Fig.7.16 shows the different meshes obtained due to uniform and adaptive refinement respectively. Fig.7.14 shows a plot of relative error norm *vs* number of dof for a Bracket example. It is seen from the graph that initial coarse mesh having 108 dof, has a error of 49.7%. We can observe from the graph that the criteria of $\eta < \eta_{all}$(15%in this case) is achieved at 3070 dof with adaptive refinement, while even at 3900 dof uniformly refined mesh possesses error of 17.8%, which is greater than $\eta_{all}$. The stress plots for the bracket problem are shown in Fig.7.17.

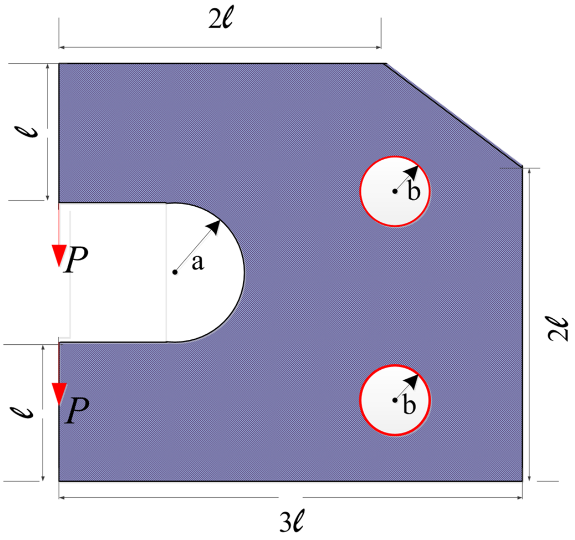


Figure 7.13: Mechanical connection.

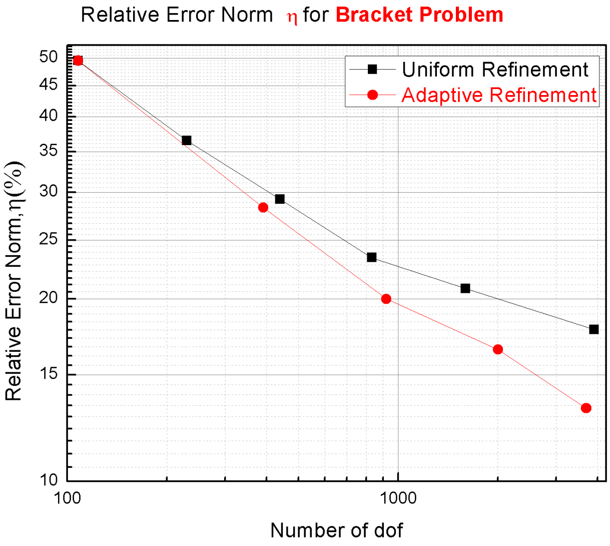


Figure 7.14: Plot of relative error norm $\eta$ for Bracket problem.

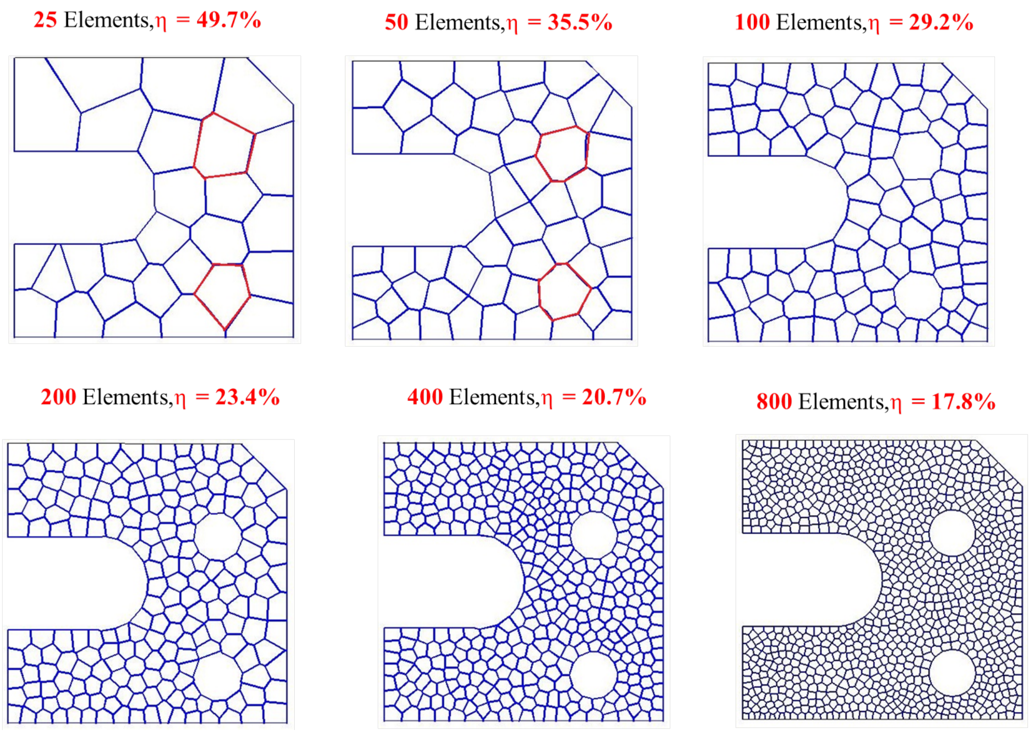


Figure 7.15: Meshes generated due to uniform refinement of the bracket.

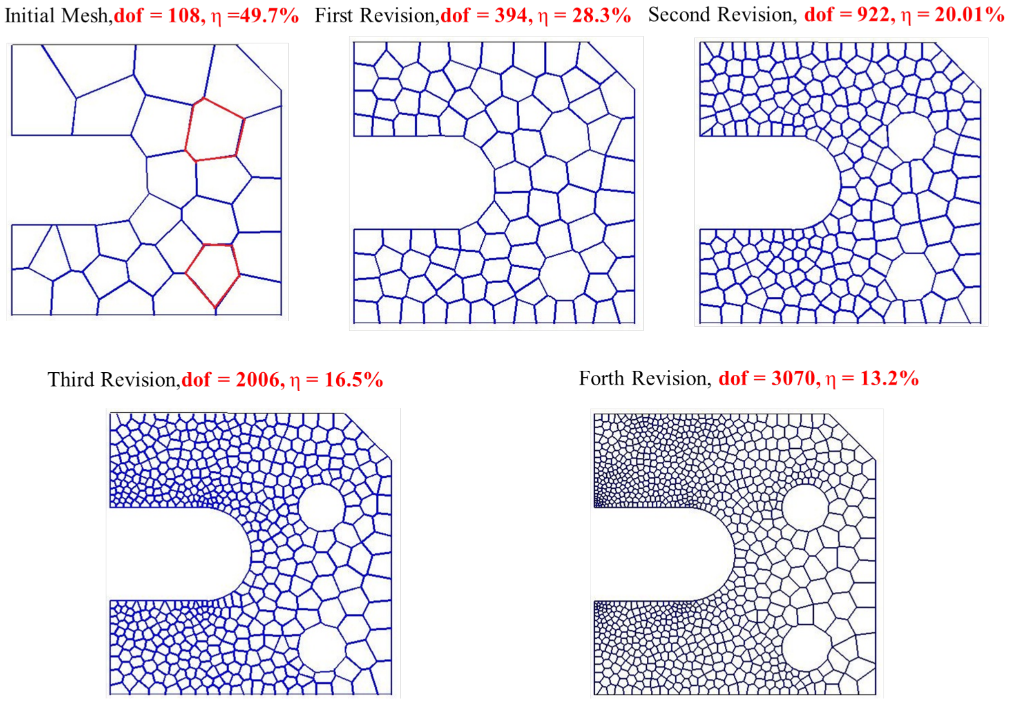

Initial Mesh,dof = 108, η =49.7%
First Revision,dof = 394, η = 28.3%
Second Revision, dof = 922, η = 20.01%
Third Revision,dof = 2006, η = 16.5%
Forth Revision, dof = 3070, η = 13.2%

Figure 7.16: Meshes generated due to adaptive refinement of the bracket

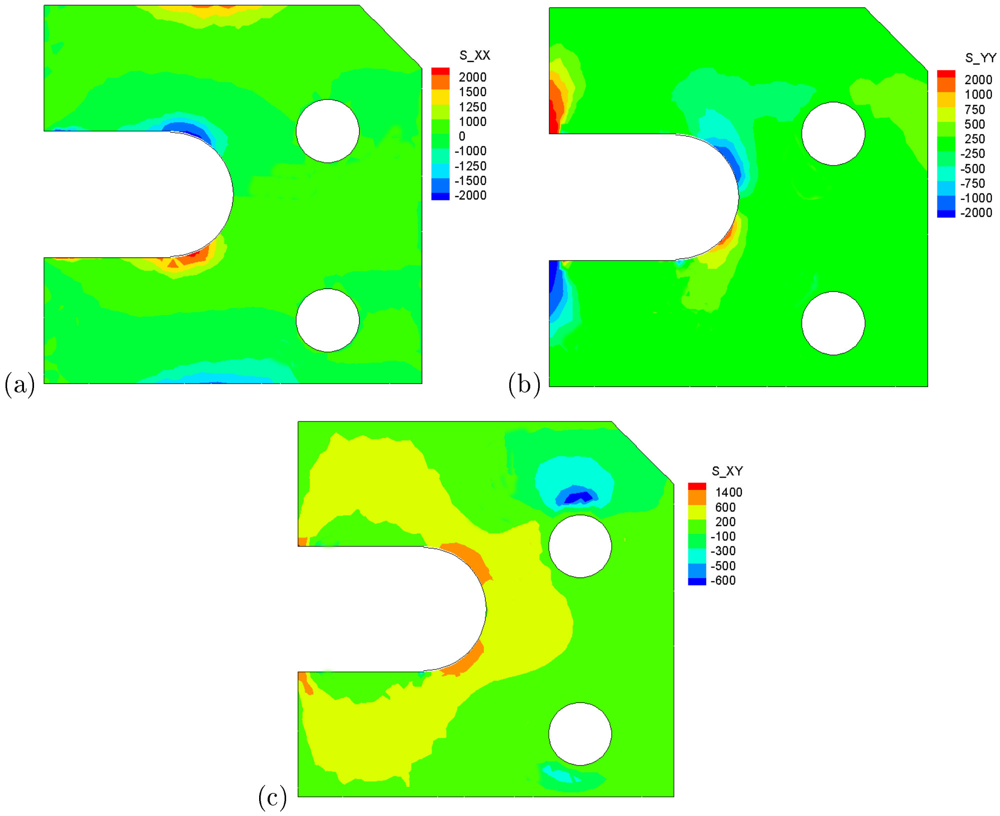


Figure 7.17: Stress plots plot for Bracket Problem.

# List of Figures

# List of Tables

# References

[1] K. Y. Sze and N. Sheng. Polygonal finite element method for nonlinear constitutive modeling of polycrystalline ferroelectrics. *Finite Elements in Analysis and Design* 42, (2005) 107–129.

[2] A. Jayabal, K. Menzel and A. Arockiarajan. Micromechanical modelling of switching phenomena in polycrystalline piezoceramics: application of polygonal finite element approach. *Computational Mechanics* 48, (2011) 421–435.

[3] S. W. Dohrmann, C. R. Key and M. W. Heinstein. A method for connecting dissimilar finite element meshes in two dimensions. *International Journal for Numerical Methods in Engineering* 48, (2000) 655–678.

[4] Y. A. Kuznetsov. Mixed finite element method for diffusion equations on polygonal meshes with mixed cells. *Journal of Numerical Mathematics* 14, (2006) 305–315.

[5] M. M. Rashid and P. M. Gullett. On a finite element method with variable element topology. *Computatioanl Methods in Applied Mechanics and Engineering* 190, (2000) 1509–1527.

[6] K. Y. Dai, G. R. Liu, and T. T. Nguyen. An n-sided polygonal smoothed finite element method (nSFEM) for solid mechanics. *Finite Elements in Analysis and Design* 43, (2007) 847–860.

[7] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. Polygonal finite element for topology optimization: A unifying paradigm. *International Journal for Numerical Methods in Engineering* 82, (2007) 671–698.

[8] B. Lin and S. N. Chandlerwilde. Numerical conformal mapping and mesh generation for polygonal and multiply connected regions. *Journal of Hydroinformatics* 2, (2000) 255–266.

[9] T. Nguyen-Thoi, G. R. Liu, and H. Nguyen-Xuan. An n-sided polygonal edge - based smoothed finite element method(nES-FEM) for solid mechanics. *International Journal for Numerical methods in Biomedical Engineering* 27, (2010) 1446–1472.

[10] M. Kraus, A. Rajagopal, and P. Steinmann. Investigations on polygonal finite element method: Constrained adaptive Delaunay tessellation and conformal interpolants. *Computers and Structures.* 120, (2013) 33–46.

[11] S. Natarajan, S. Bordas, and D. R. Mahapathra. Numerical integration over arbitrary polygonal domains based on Schwarz-Christoffel conformal mapping. *International Journal for Numerical Methods in Engineering* 80, (2009) 103–134.

[12] S. Natarajan, S. Bordas, and D. R. Mahapathra. Integrating strong and weak discontinuities without integration sub cells and example applications in XFEM/GFEM framework. *International Journal for Numerical Methods in Engineering* 83, (2010) 269–294.

[13] C. J. Li, P. Lamberti, and C. Dagnino. Numerical integration over polygons using an eight-node quadrilateral spline finite element. *Journal of Computational and Applied Mathematics* 233, (2009) 279–292.

[14] S. E. Mousavi, H. Xiao, and N. Sukumar. Generalized Gaussian quadrature rules on arbitrary polygons. *International Journal of Numerical Methods in Engineering* 82, (2009) 99–113.

[15] J. Yvonnet, D. Ryckelynck, P. Lorong, and F. Chinesta. A new extension of the natural element method for non-convex and discontinuous problems: the constrained natural element method (C-NEM). *International Journal for Numerical Methods in Engineering* 60, (2004) 1451–1474.

[16] I. Babuska and A. Aziz. The Mathematical Foundation of the Finite Element Method with Applications to Partial Differential Equations,. *Academic Press, New York,* .

[17] I. Babuska and W. C. Rheniboldt. A-posteriori error estimates for the finite element method,. *International Journal for Numerical Methods in Engineering* 12, (1978) 1597–1615.

[18] I. Babuska and A. Miller. A feedback finite element method with a posteriori error estimation: Part I. The finite element method and some basic properties of the a posteriori error estimator,. *Computer Methods in Applied Mechanics and Engineering,* 61, (1987) 1–40.

[19] O. C. Zienkiewcz and E. Rank. A simple error estimator in finite element method. *Communications in Applied Numerical Methods* 3, (1987) 243–249.

[20] E. Stein, R. Marcus, and O. Stephan. Error-controlled adaptive goal-oriented modeling and finite element approximations in elasticity. *Computer Methods in Applied Mechanics and Engineering* 196, (2007) 3598–3613.

[21] J. T. Oden and S. Pruhomme. Goal-oriented Error Estimation and Adaptivity for the Finite Element Method. *International Journal of Computers and mathematics* 41, (1999) 735–756.

[22] P. M. Mohite and C. S. Upadhyay. Focussed Adaptivity for laminate plates. *International Journal of Computers and Structures* 81, (2003) 287–298.

[23] A. Rajagopal and S. Sivakumar. An r-h adaptive strategy for analysis of plane problems with bimaterial interfaces. *Computational Mechanics* 41, (2007) 49–72.

[24] J. Mackerle. Error estimates and adaptive finite element methods- A bibliography (1990-2000). *Engineering Computations* 18(5/6), (2001) 802–914.

[25] J. O. Dow. A unified approach to finite element method and error analysis procedure,. *Academic Press, New York,* .

[26] R. Denzer, M. Scherer, and P. Steinmann. An adaptive singular finite element in nonlinear fracture mechanics. *Defect and Material Mechanics, C Dascalu, G A Maugin, C Stolz (eds)* 181–190.

[27] J. Friederich, G. Leugering, and P. Steinmann. Adaptive refinement based on asymptotic expansions of finite element solution for node insertion in 1D. *GAMM-Mitteilungen* 35, (2012) 175–190.

[28] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. Polymesher: a general purpose mesh generator for polygonal elements in Matlab. *Structural and Multidisciplinary Optimization* 45, (2012) 309–328.

[29] C. Talischi, G. H. Paulino, A. Pereira, and I. F. M. Menezes. PolyTop: A Matlab implementation of a general topology optimization framework using unstructured polygonal finite element meshes. *Structural and Multidisciplinary Optimization* .

[30] V. V. Belikov, V. D. Ivanov, V. K. Kontorovich, S. A. Korytnik, and A. Y. Semenov. The non-Sibsons interpolation: A new method of interpolation of the values of a function on an arbitrary set of points. *Computational Mathematics and Mathematical Physics* 37, (1997) 9–15.

[31] N. H. Christ, R. Friedberg, and T. D. Lee. Weights of links and plaquettes in a random lattice. *Nuclear Physics B* 210, (1982) 337–346.

[32] H. Hiyoshi and K. Sugihara. Two generalizations of an interpolant based on Voronoi diagrams. *International Journal of Shape Modeling* 5, (1999) 219–231.

[33] R. Sibson. A vector identity for the Dirichlet tessellation. *Mathematical Proceeding of Cambridge Philosophical Society* 87, (1980) 151–155.

[34] A. Constantiniu, P. Steinmann, T. Bobach, G. Farin, and G. Umlauf. The Adaptive Delaunay Tessellation: a neighborhood covering meshing technique. *Computational Mechanics* 42, (2008) 655–669.

[35] N. Sukumar and A. Tabarraei. Conforming polygonal finite elements. *International Journal for Numerical Methods in Engineering* 61, (2004) 2045–2066.

[36] G. Dasgupta. Integration within polygonal finite elements. *Journal of Aerospace Engineering* 16, (2003) 9–18.

[37] S. De and K. J. Bathe. The method of finite spheres. *Computational Mechanics* 25, (2000) 329–345.

[38] S. De and K. J. Bathe. The method of finite spheres with improved numerical integration. *Computers and Structures* 79, (2001) 2183–2196.

[39] M. A. Taylor. Asymmetric cubature formulas for polynomial integration in the triangle and square. *Journal of Computational and Applied Mathematics* 218, (2008) 184–191.

[40] S. Wandzura and H. Xiao. Symmetric quadrature rules on a triangle. *Computers and Mathematics with Applications* 45, (2003) 1829–1840.

[41] J. Ma, V. Rokhlin, and S. Wandzura. Generalized quadrature of systems of arbitrary functions. *Society of Industrial and Applied Mathematics* 33, (1996) 971–996.

[42] P. Silvester. Symmetric quadrature formula for simplexes. *Mathematics of Computation* 24, (1970) 95–100.

[43] J. N. Lyness and D. Jespersen. Moderate degree symmetric quadrature rule for triangles. *IMA Journal of Applied Mathematics* 15, (1975) 19–32.

[44] H. Xiao and Z. Gimbutas. A numerical algorithm for construction of efficient quadrature rules in two and higher dimensions. *Computers and Mathematics with Applications* 59, (2010) 663–676.

[45] E. Hinton and J. S. Campbell. local and global smoothing of discontinuous finite element functions using a least squares method. *International Journal for Numerical Methods in Engineering* 8, (1974) 461–480.

[46] J. T. Oden and H. J. Brauchil. On the calculation of consistent stress distributions in finite element approximations. *International Journal for Numerical methods in Engineering* 3, (1971) 317–325.