A Novel Polygonal Finite Element Method: Virtual Node Method

X. H. Tang[a,*], C. Zheng[a], and J. H. Zhang[a]

[a] *State Key Lab of Hydraulics and Mountain River Engineering, Sichuan University, Chengdu, P. R. China(*X.H.Tang84@Gmail.com)*

Abstract. Polygonal finite element method (PFEM), which can construct shape functions on polygonal elements, provides greater flexibility in mesh generation. However, the non-polynomial form of traditional PFEM, such as Wachspress method and Mean Value method, leads to inexact numerical integration. Since the integration technique for non-polynomial functions is immature. To overcome this shortcoming, a great number of integration points have to be used to obtain sufficiently exact results, which increases computational cost. In this paper, a novel polygonal finite element method is proposed and called as virtual node method (VNM). The features of present method can be list as: (1) It is a PFEM with polynomial form. Thereby, Hammer integral and Gauss integral can be naturally used to obtain exact numerical integration; (2) Shape functions of VNM satisfy all the requirements of finite element method. To test the performance of VNM, intensive numerical tests are carried out. It found that, in standard patch test, VNM can achieve significantly better results than Wachspress method and Mean Value method. Moreover, it is observed that VNM can achieve better results than triangular 3-node elements in the accuracy test.

Keywords: virtual node method; polygonal finite element method; partition of unity; accuracy
PACS: 87.16.Ac; 87.17.Aa

INTRODUCTION

In 1975, Wachspress used the concepts of projective geometry to develop rational interpolant for convex polygons [1]. In 2003, Floater [2] proposed mean value method and extended the polygonal finite element method to all polygonal elements. Polygonal finite element method (PFEM), which can construct shape functions on polygonal element with n nodes ($n\geq 3$), are potentially useful in many areas, such as modeling of polycrystalline [3] and serving as transitional elements in adaptive computations [4].

However, the non-polynomial form of traditional polygonal finite element method, such as Wachspress method and mean value method, leads to inexact numerical integration. Since the integration technique for non-polynomial is immature. Hammer integral and Gauss integral, which are only precise for polynomial functions, are still widely used for the integration of PFEM [5]. In this paper, a novel PFEM with polynomial form are proposed and named as virtual node method (VNM). With polynomial form, Hammer integral and Gauss integral can be naturally used for VNM to achieve exact numerical integral.

FORMULATION FOR VIRTUAL NODE METHOD

Consider a polygonal domain Ω in the 2D which is described by a set of scattered nodes $\{P_1, P_2,\ldots, P_n\}$, as shown in Fig. 1 n is the number of nodes. P_i denotes node i with co-ordinates $\mathbf{x}_i=(x_i, y_i)$. Firstly, using centroid of element, the polygonal domain Ω is divided into a set of virtual triangles $\{T_1, T_2,\ldots,T_n\}$, as shown in Fig. 2. It should be noticed that VNM can not be used to construct shape function for polygonal elements, whose centroids are not in polygonal domain. Here, the node $P_k^{'}$, which is at the centroid of element, is a virtual node.

Introduce an insert point P with coordinate $\mathbf{x}\equiv(x,y)$, The approximation function is conformed as:

CP1233 *The 2nd International ISCM Symposium and The 12th International EPMESC Conference*
J. W. Z. Lu, A.Y.T. Leung, V. P. Iu, and K. M. Mok

$$u^h(\mathbf{x}) = \hat{R}^{[I]}(\mathbf{x})\hat{u}^{[I]}(\mathbf{x}) + \tilde{R}^{[I]}(\mathbf{x})\tilde{u}(\mathbf{x}) \tag{1}$$

where the superscript [*I*] indicates that insert point *P* is in virtual triangle T_I. $\tilde{u}(\mathbf{x})$ is least-square approximation and $\hat{u}^{[I]}(\mathbf{x})$ is 3-node approximation. The formulas for least-square approximation and 3-node approximation are introduced in section 2.1 and section 2.2, respectively. $\hat{R}^{[I]}(\mathbf{x})$ and $\tilde{R}^{[I]}(\mathbf{x})$ are weight functions of $\hat{u}^{[I]}(\mathbf{x})$ and $\tilde{u}(\mathbf{x})$, respectively, which are introduced in section 2.3 in detail.

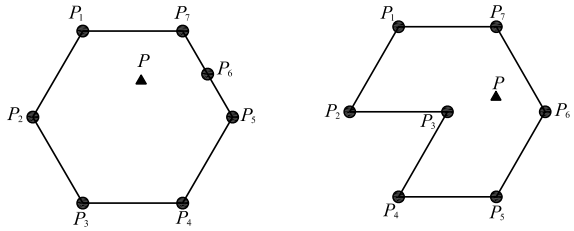
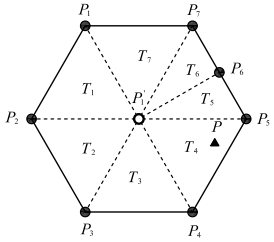


FIGURE 1. Polygonal elements



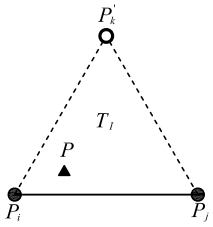


FIGURE 2. Virtual triangles

2.1 Least-Square Approximation

The least-square method is widely used [6] and can be represented as:

$$\tilde{u}(\mathbf{x}) = \sum_{i=1}^{n} \tilde{\Phi}_i(\mathbf{x}) a_i = \tilde{\mathbf{\Phi}}^{\mathrm{T}} \mathbf{a} \tag{2}$$

$$\tilde{\mathbf{\Phi}}^{\mathrm{T}} = \mathbf{p}^{\mathrm{T}}(\mathbf{x})\mathbf{A}^{-1}\mathbf{B} = \left\{ \tilde{\Phi}_1(\mathbf{x}) \quad \tilde{\Phi}_2(\mathbf{x}) \quad \cdots \quad \tilde{\Phi}_n(\mathbf{x}) \right\} \tag{3}$$

$$\mathbf{a} = (a_1, a_2, \cdots, a_n)^{\mathrm{T}} \tag{4}$$

Where a_i is nodal displacement associated with node *i*. $\mathbf{p}(\mathbf{x})$ is a vector of basis functions built by the Pascal's triangle and *n* is the number of nodes of polygonal element. For 2D, basis function $\mathbf{p}(x)$ of order 2 is written by:

$$\mathbf{p}^{\mathrm{T}}(\mathbf{x}) = \left\{ p_1(\mathbf{x}) \quad p_2(\mathbf{x}) \quad \cdots \quad p_6(\mathbf{x}) \right\} = \left\{ 1 \quad x \quad y \quad xy \quad x^2 \quad y^2 \right\} \tag{5}$$

The moment matrix $\mathbf{A}$ and basis matrix $\mathbf{B}$ are expressed respectively:

$$\mathbf{A} = \sum_{i=1}^{n} \mathbf{p}(\mathbf{x}_i)\mathbf{p}^{\mathrm{T}}(\mathbf{x}_i), \ \mathbf{B} = \left[\mathbf{p}(\mathbf{x}_1) \quad \mathbf{p}(\mathbf{x}_2) \quad \cdots \quad \mathbf{p}(\mathbf{x}_n) \right] \tag{6}$$

2.2 3-Node Approximation

Similar to conventional 3-node finite element method, $\hat{u}^{[I]}(\mathbf{x})$ is constructed as:

$$\hat{u}^{[I]}(\mathbf{x}) = \hat{\Phi}_i^{[I]}(\mathbf{x})a_i + \hat{\Phi}_j^{[I]}(\mathbf{x})a_j + \hat{\Phi}_k^{[I]}(\mathbf{x})a_k^{'} \tag{7}$$

$$\hat{\Phi}_i^{[I]}(\mathbf{x}) = (x(y_j - y_k) + y(x_k - x_j) + (x_j y_k - x_k y_j))/A \tag{8}$$

$$\hat{\Phi}_j^{[I]}(\mathbf{x}) = (x(y_k - y_i) + y(x_i - x_k) + (x_k y_i - x_i y_k))/A \tag{9}$$

$$\hat{\Phi}_k^{[I]}(\mathbf{x}) = (x(y_i - y_j) + y(x_j - x_i) + (x_i y_j - x_j y_i))/A \tag{10}$$

$$A = x_i y_j + x_j y_k + x_k y_i - y_i x_j - y_j x_k - y_k x_i \tag{11}$$

Note that, $a_k^{'}$ is the virtual nodal displacement, which is equal to the value of least-square approximation $\tilde{u}(\mathbf{x})$ at virtual node $P_k^{'}$:

$$a_k^{'} = \tilde{u}(\mathbf{x}_k) = \sum_{l=1}^{n} \tilde{\Phi}_l(\mathbf{x}_k) a_l \tag{12}$$

Substitute Eq.(12) to Eq.(7), the 3-node approximation is manipulated as:

$$\hat{u}^{[I]}(\mathbf{x}) = \hat{\Phi}_i^{[I]}(\mathbf{x})a_i + \hat{\Phi}_j^{[I]}(\mathbf{x})a_j + \hat{\Phi}_k^{[I]}(\mathbf{x}) \sum_{l=1}^{n} \tilde{\Phi}_l(\mathbf{x}_k) a_l \tag{13}$$

2.3 WeightFunctions

Weight functions $\hat{R}^{[I]}(\mathbf{x})$ and $\tilde{R}^{[I]}(\mathbf{x})$ can be represented as：

$$\hat{R}^{[I]}(\mathbf{x}) = \hat{\Phi}_i^{[I]}(\mathbf{x}) + \hat{\Phi}_j^{[I]}(\mathbf{x}) = (x(y_j - y_i) + y(x_i - x_j) + (x_j y_k - x_k y_j + x_k y_i - x_i y_k))/A \tag{14}$$

$$\tilde{R}^{[I]}(\mathbf{x}) = \hat{\Phi}_k^{[I]}(\mathbf{x}) = (x(y_i - y_j) + y(x_j - x_i) + (x_i y_j - x_j y_i))/A \tag{15}$$

Obviously, weight functions $\hat{R}^{[I]}(\mathbf{x})$ and $\tilde{R}^{[I]}(\mathbf{x})$ satisfy following properties:

(1) Partition of unit, $\hat{R}^{[I]}(\mathbf{x}) + \tilde{R}^{[I]}(\mathbf{x}) \equiv 1$

(2) On the boundaries of polygonal elements, $\hat{R}^{[I]}(\mathbf{x}) = 1$ and $\tilde{R}^{[I]}(\mathbf{x}) = 0$.

(3) At the virtual node P'_k, $\hat{R}^{[I]}(\mathbf{x}) = 0$ and $\tilde{R}^{[I]}(\mathbf{x}) = 1$.

2.4 Formulation for shape functions

Approximation function $u^h(\mathbf{x})$ can be represented in general form:

$$u^h(\mathbf{x}) = \sum_{l=1}^{n} N_l(\mathbf{x}) a_l \tag{16}$$

Where $N_l(\mathbf{x})$ is the shape function associated with node l in virtual triangle domain T_I, and can be represented as:

$$N_l(\mathbf{x}) = \hat{R}^{[I]}(\mathbf{x})\left((\delta_{il} + \delta_{jl})\hat{\Phi}_l^{[I]}(\mathbf{x}) + \hat{\Phi}_k^{[I]}(\mathbf{x})\tilde{\Phi}_l(\mathbf{x}_k)\right) + \tilde{R}(\mathbf{x})\tilde{\Phi}_l(\mathbf{x}) \tag{17}$$

Where

$$\begin{cases} \delta_{ml} = 0 & m \neq l \\ \delta_{ml} = 1 & m = l \end{cases} \tag{18}$$

Virtual node method satisfy following properties:

(1) $u^h(\mathbf{x})$ is piece-wise linear (C^0 function) on the boundaries of domain Ω:

$$u^h(t) = tu_1 + (1-t)u_2, \mathbf{x} \in \partial\Omega, t \in [0,1] \tag{19}$$

(2) Delta property

$$\begin{cases} N_i(\mathbf{x}_j) = 0 & i \neq j \\ N_i(\mathbf{x}_j) = 1 & i = j \end{cases} \tag{20}$$

(3) Linear completeness

$$u^h(\mathbf{x}) = \sum_{i=1}^{n} \mathbf{x}_i N_i(\mathbf{x}) = \mathbf{x} \tag{21}$$

(4) u^h(x) is C^∞ inside each triangular elements and is C^0 on the junction between two virtual triangles.

Any smooth function which satisfies the requirements above can be used to construct a shape functions of finite element method [7]. As shown in Fig.3, it is easily found that shape functions which are constructed by VNM, is piece-wise linear on the boundaries of domain Ω and satisfy Delta property.

node 6 node 7

FIGURE 3. Shape functions for VNM

NUMERICAL TESTS

Several numerical examples are studied to test the performance of VNM. In the following tests, Hammer integration with *nsp*-point integraion rule on each virtual triangular element is used to compute the last integral. To

examine the accuracy and convergence, the relative L^2 error in displacement norm and in energy norm are defined respectively as follows:

$$e_d = \sqrt{\int_\Omega (\mathbf{u}^{\text{exact}} - \mathbf{u}^{\text{numerical}})^2 \, \mathrm{d}\Omega \Big/ \int_\Omega (\mathbf{u}^{\text{exact}})^2 \, \mathrm{d}\Omega} \tag{22}$$

$$e_e = \sqrt{\int_\Omega (\boldsymbol{\varepsilon}^{\text{exact}} - \boldsymbol{\varepsilon}^{\text{numerical}})^{\mathrm{T}} \mathbf{D} (\boldsymbol{\varepsilon}^{\text{exact}} - \boldsymbol{\varepsilon}^{\text{numerical}}) \mathrm{d}\Omega \Big/ \int_\Omega (\boldsymbol{\varepsilon}^{\text{exact}})^{\mathrm{T}} \mathbf{D} \boldsymbol{\varepsilon}^{\text{exact}} \mathrm{d}\Omega} \tag{23}$$

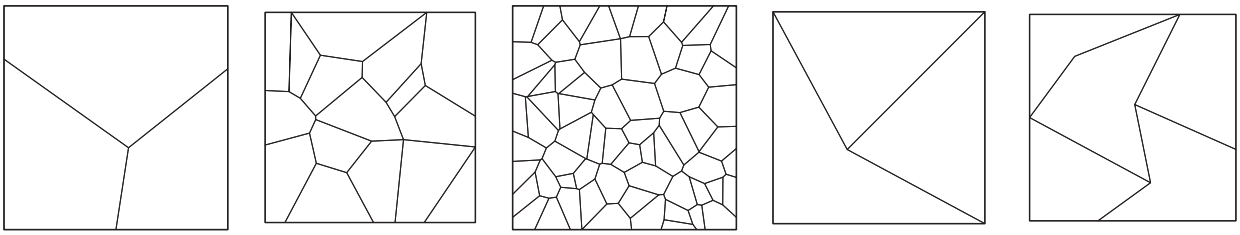
where the superscript "exact" represents the exact or analytical solution and the "numerical" denotes a numerical solution obtained using a numerical technique. The variable units used in the work are based on international standard unit system unless specially denoted.

3.1 Standard Patch Test

The problem is studied in a 1×1 square domain, and the displacements are prescribed on all outside boundaries by the following linear function

$$u_x = x \, , u_y = y \tag{24}$$

Firstly, three convex meshes are concerned. The number of elements (nodes) are 3(8), 16(30) and 64(129) respectively, as shown in Fig. 4 (A-C). Secondly, the performance on the concave meshes is studied with two different meshes and the number of elements (nodes) are 3(5) and 5(11) respectively, as shown in Fig. 4 (D,E). In Table 1, the results are shown for *nsp*=3 and *nps*=16 integration rule within each virtual triangle respectively. Relative $L^2(\Omega)$ error in displacement norm results of VNM is indicate that their accuracy of displacement in the standard patch test is 10^{-10}, which is better that that of Wachspress method and mean value method (10^{-6}).



(a) mesh A (b) mesh B (c) mesh C (a) mesh D (b) mesh E

FIGURE 4. Polygonal meshes for standard patch test

TABLE 1. Relative L^2 in displacement norm for the standard patch test.

Mesh	nps	Wachspress	Mean Value	Virtual Nnode Method
A	3	7.7888E-004	7.6959E-004	5.6446E-009
	16	3.2112E-006	8.2247E-005	1.1321E-009
B	3	2.3807E-003	8.0739E-004	1.0532E-008
	16	2.1809E-004	7.6165E-005	5.1493E-010
C	3	5.2035E-003	1.4244E-003	1.4935E-008
	16	6.4723E-004	1.7329E-004	2.2789E-010
D	3		1.3246E-004	1.4169E-009
	16		5.7134E-005	4.4399E-010
E	3		2.1756E-003	2.8503E-009
	16		1.6865E-004	9.5130E-010

3.2 Cantilever Beam

A 2D cantilever beam with length *L*, height *D* and unit thickness is studied for the various behavior of VNM. The beam is fixed at the left end and subjected to a parabolic traction *P* on the right edge of the beam, as shown in Fig. 5. The analytical solutions of displacement and stress [8] for the plane stress case are given by:

$$u_x = -P\left(y - 0.5D\right)\left[(6L - 3x)x + (2+\nu)(y^2 - Dy)\right] \Big/ (6EI) \tag{25}$$

$$u_y = P\left[3\nu\left(y - 0.5D\right)^2 (L - x) + 0.25(4 + 5\nu)D^2 x + (3L - x)x^2\right] \Big/ (6EI) \tag{26}$$

$$\sigma_{xx} = -P(L-x)\big(y-0.5D\big)/I\ ,\ \sigma_{yy} = 0\ ,\ \ \sigma_{xy} = -Py(y-D)/2I \tag{27}$$

where I is the moment of inertia given as $I = D^3/12$. The geometric parameters in the computation are taken as: L=8, D=1, P=1, and the material parameters are taken as E=3×10^7 and v=0.25, and the plane strain condition is assumed. In the computation, the nodes on the boundary of x=0 are constrained using the exact displacements given by Eq. (25) and Eq.(26), and the traction are specified on the boundary of x=L using Eq. (27)

Five discrete models with polygonal elements are concerned and the numbers of elements (nodes) are 166(82), 283(568), 1047(1936), 3669(7340), 4096(6417), as shown in Fig. 6 (a). To compare with standard FEM, three different discrete models with triangular elements are concerned. The numbers of elements(nodes) are : 264(169), 1056(601), 17622 (9112), as shown in Fig. 7 (b). The convergence curves are plotted in Fig. 8 for standard FEM and VNM respectively. Obviously, the solution of virtual node method is significantly better than that of standard FEM.

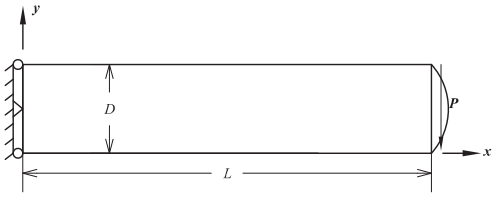
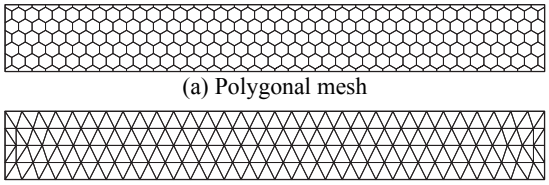


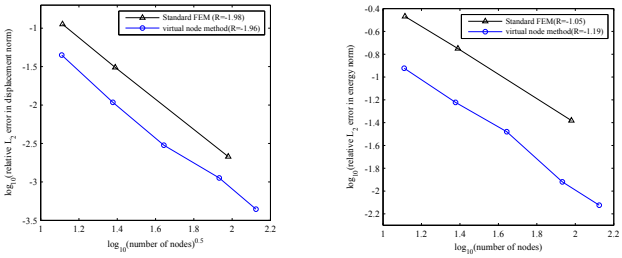
FIGURE 5. A 2D cantilever beam subjected to a parabolic traction on the right end



(a) Polygonal mesh

(b) Triangular mesh

FIGURE 6. Meshes for cantilever beam problem



(a) Relative error in displacement norm

(b) Relative error in energy norm

FIGURE 7. Comparison of accuracy for cantilever beam problem

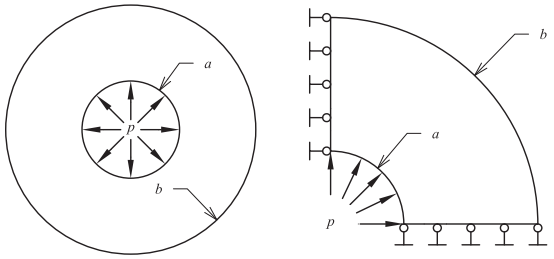
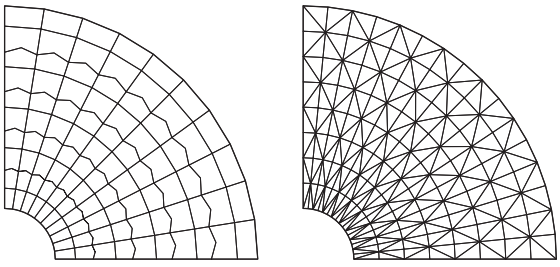


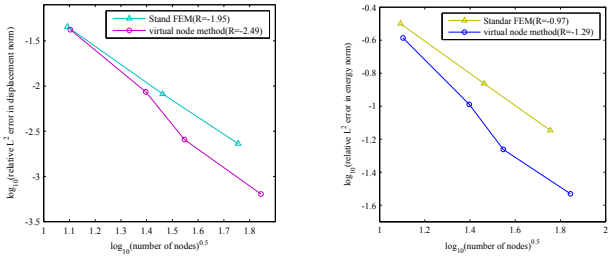
FIGURE 8. Hollow cylinder subjected to internal pressure, and its quarter model



(a) Concave mesh

(b)Triangular mesh

FIGURE 9. Mesh for Hollow cylinder problem



(a) Relative error in displacement norm

(b) Relative error in energy norm

FIGURE 10. Comparison of accuracy on concave mesh for Hollow cylinder problem

3.3 Hollow Cylinder under Internal Pressure

A hollow cylinder with an internal radius a, an external radius b and unit thickness is considered as another typical problems to validate the virtual node method. As shown in Fig. 9, the uniform pressure p=1 is applied to the inner surface (r =a), while the outer surface (r =b) is free of traction. Due to the symmetry of the problem, only one-quarter of the cylinder is modeled. The analytical solution is available in [9].

$$u_r(r) = a^2 pr\Big[1-v+b^2(1+v)/r^2\Big]\Big/\Big[E(b^2-a^2)\Big],\ u_\theta = 0 \tag{28}$$

$$\sigma_r(r) = a^2 p(1-b^2/r^2)/(b^2-a^2)\ ,\ \sigma_\theta(r) = a^2 p(1+b^2/r^2)/(b^2-a^2)\ ,\ \sigma_{r\theta} = 0 \tag{29}$$

In the numerical computations, the following parameters are chosen: a=1, b=5, p=1, and the plane stress condition is assumed. The material used is linear elastic with Young's modules $E = 1\times10^3$ and ν=0.25.

In this convergence study, four different meshes with polygonal elements are constructed with the different numbers of elements (nodes) are : 100(161), 400(621), 800(1241), 3200 (4881), as shown in Fig. 10 (a). Meanwhile, the meshes with triangular elements are concerned, the numbers of elements (nodes) are: 256(153), 1560(840), 6240(3239), as shown in Fig.9 (b). In Fig. 11, the relative L^2 error in displacement and energy norms are plotted. Similar to the performance in cantilever beam, the solution of VNM is significantly better than that of standard FEM.

CONCLUSION

In this paper, the formulation for a novel virtual node method is introduced. The present method is attractive due to the following reasons:

(1) Virtual node method is used in solid mechanics problem and it is observed that virtual node method achieves good results on polygonal elements.

(2) Since the approximation function of virtual node method is polynomial form, Gaussian integral and Hammer integral can be used for virtual nodes method to obtain exact results. However, the approximation function of traditional PFEM, such as Wachspress method and mean value method, is non-polynomial and it leads to excessive error of numerical integration.

(3) In standard patch test, the relative L^2 error in displacement norm results indicates that the accuracy of virtual node method in the patch test is 10^{-10}. It is significantly better than that of Wachspress method and mean value, whose accuracy is 10^{-4}.

(4) It is found in convergence study that the results of virtual node method is better than that of traditional triangular 3-node elements

It should be noted that VNM can not be used to construct shape function for polygonal elements, whose centroids are not in polygonal domain.

REFERENCES

1. E.L. Wachspress, *A Rational Finite Element Basis*, Academic Press, New York. 1975.
2. M.S. Floater, "Mean value coordinates", *Computer Aided Geometric Design*, 2003, 20(1), pp. 19-27.
3. S. Ghosh, S. Moorthy."Elastie-Plastic analysis of arbitray heterogeneous materials with the Voronoi cell finite element method", *Computer Methods Applied in Mechanics and Engineering*, 1995, 121, pp. 373-409.
4. A. Tabarraei, N. Sukumar, "Adaptive computations on conforming quadtree meshes", *Finite Element in Analysis and Design*, 2005, 41, pp. 686-702.
5. N. Sukumar, E.A. Malsch, "Recent advances in the construction of polygonal finite element interpolants", Archives of Computational Methods in Engineering, 2006, 13(1), pp.129-163.
6. J. Stoer, *Numerical analysis and scientific computation*, New York, 1998.
7. G. Strang and G. Fix, *An Analysis of the Finite Element Method*. Pretice-Hall, Engle-wood Cliffs, N.J. 1973.
8. S. P. Timoshenko, J. N. Goodier, *Theory of Elasticity*, third edition, McGraw, New York, 1970.
9. R. J. Roark, W. C. Young, *Formulas for stress and strain*. McGraw, New York, 1975.