# “Plane Stress and Plane Strain with the Finite Element Method - Billinear and Triangular Elements”

**Luis Fernando Verduzco Martínez [1]**

[1] luiz.verduz@gmail.com

Faculty of Engineering, Autonomous University of Querétaro, Santiago de Querétaro, Mexico

### Overview

This software is the implementation of the Finite Element Method for the Plane Stress and Plain Strain analysis of a Submarine Concrete Tunnel subject to hydrostatic pressures. The mesh and topology is generated with the **function** *TunnelMeshGen* either of four-node-billinear elements or of three-node-triangular elements. Numerical integration with the *Gaussian Quadrature* is used, in which parent unitary finite elements are considered. Both CALFEM functions and Built-in MatLab functions are used to plot the deformed and un-deformed meshes once the analysis are finished.

**Keywords: Finite Element Method, Plane Stress, Plane Strain, Gaussian Quadrature, Billinear Elements, Triangular Elements**

# 1 Numerical Integration

## 1.1 Gaussian Quadrature

All the Newton-Cotes formulas use values of the function at equally spaced points. Nevertheless, this restriction can significantly decrease the accuracy of the approximation. For instance when the trapezoidal rule for the integration of the function in **Fig. 1** is executed by taking the endpoints of the graph of the function, less accurate results are obtained than when the integration is made by taking certain points within the range of analysis as in **Fig. 2**:

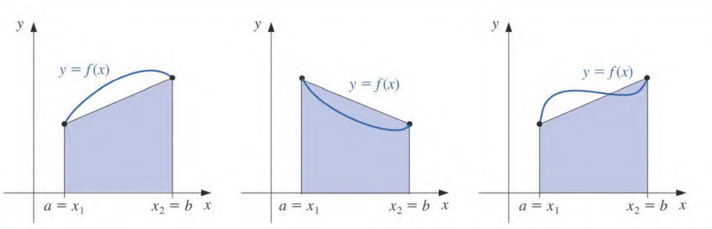

**Figure 1.** Integration of a function with the trapezoidal by the endpoints.

Therefore, an optimal formulation could be formalized so that such point within the endpoints could be determined to encounter the best approximation to the integral. Gauss developed precisely this approach called the *Gaussian Quadrature* in 1814 in which some coefficients $c_1, c_2, ...c_n$ and nodes $x_1, x_2, ..., x_n$ are chosen to minimize the expected error for the approximation as (1):

$$\int_a^b f(x)dx \approx \sum_{i=1}^{n} c_i f(x_i) \tag{1}$$

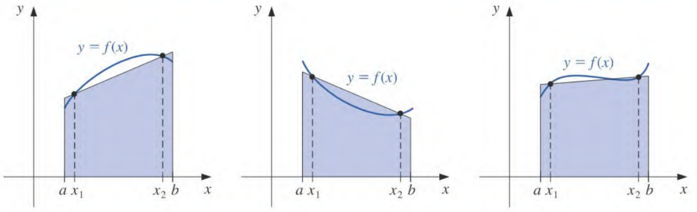

**Figure 2.** Integration of a function with the trapezoidal by points within the endpoints.

The largest class of polynomials for which it is reasonable to expect this formula to be exact is that of degree $2n - 1$ for which the $n$ coefficients and $n$ nodes apply.

Now, in order to stablish a general formula to determine the position of such nodes and their respective coefficients the Legendre polynomial must be considered (see [2]), in which it is supposed that $x_1, x_2, ..., x_n$ are the roots of a $n$th Legendre polynomial $P_n(x)$ and that for each $i = 1, 2, ..., n$, the numbers $c_i$ are defined by:

$$c_i = \int_{-1}^{1} \prod_{j=1 j \neq i}^{n} \frac{x - x_j}{x_i - x_j} dx \tag{2}$$

$$\int_{-1}^{1} P(x) dx = \sum_{i=1}^{n} c_i P(x_i) \tag{3}$$

Hence, the constants $c_i$ needed for the quadrature rule can be generated from the Legendre polynomials which are shown next in **Table 1** (also known as *Gauss points*) for a max of $n = 3$:

**Table 1.** Gauss points and weight functions to apply Gaussian quadrature.

| $n$ | $\xi_i$ | $W_i$ |
|---|---|---|
| 1 | 0 | 2 |
| 2 | $\pm\frac{1}{\sqrt{3}}$ | 1 |
| 3 | $\pm 0.774596692$ | 0.55555555 |
|  | 0 | 0.88888888 |

Now, in order for the Gaussian Quadrature to apply for an arbitrary interval other than $[-1, 1]$ a change of variables must take place as (4) - see **Fig. 3**

$$x = \frac{1}{2}[(b - a)t + a + b] \tag{4}$$

$$dx = \frac{1}{2}(b - a)dt \tag{5}$$

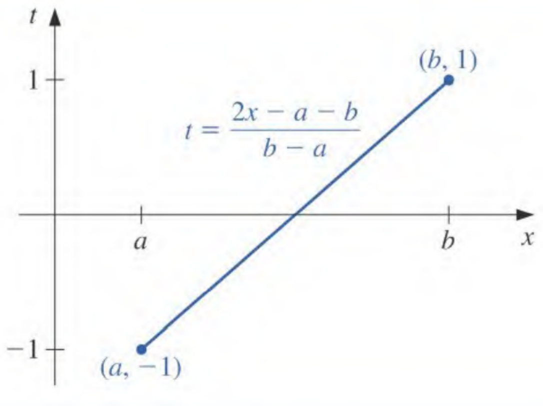

**Figure 3.** Scaling line to transform the space $[-1, 1]$ to $[a, b]$ and viceversa.

This way, the Gaussian quadrature can be applied to any interval $[a, b]$ as (6):

$$\int_a^b f(x)dx = \int_{-1}^1 f(x)\frac{(b-a)}{2}dx \tag{6}$$

### 1.1.1 Multiple Integrals

For multiple integrals the Gaussian quadrature could be easily adapted. For instance, for a double integral, the Gaussian quadrature could be applied as (7), where $x$ and $y$ are the transformed variables to their respective $[-1, 1]$ space:

$$\int_{ax}^{bx}\int_{by}^{bx} f(x,y)dydx \approx \sum_{i=1}^{n}\sum_{j=1}^{n} f(x,y)\frac{dx}{d\zeta}\frac{dy}{d\eta}cx_i cy_j \tag{7}$$

where $\chi$ and $\eta$ are defined as:

$$\frac{dx}{d\zeta} = \frac{bx - ax}{2} \tag{8}$$

$$\frac{dx}{d\eta} = \frac{by - ay}{2} \tag{9}$$

# 2 Linear Elasticity

Linear theory is derived from the assumption that the deformations that a continuum bod undergoes before the application of a set of forces and stresses are too small and, therefore, independent in time.

## 2.1 Isotropic and homogeneous materials

According to the Hooke's law for isotropic materials the constitutive equation for linear elasticity is a linear relation $D$ (constitutive matrix of materials) between the stress tensor $\sigma$ and the infinitesimal strain tensor $\epsilon$ such that equation (10) complies:

$$\sigma = D\epsilon \tag{10}$$

In which $D$ is a symmetrical matrix (as it is assumed that for isotropic materials their mechanical properties are the same for any direction of analysis), with components:

$$\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{33}\\ \sigma_{23}\\ \sigma_{13}\\ \sigma_{12}\end{bmatrix} = \begin{bmatrix}2\mu+\lambda & \lambda & \lambda & 0 & 0 & 0\\ \lambda & 2\mu+\lambda & \lambda & 0 & 0 & 0\\ \lambda & \lambda & 2\mu+\lambda & 0 & 0 & 0\\ 0 & 0 & 0 & \mu & 0 & 0\\ 0 & 0 & 0 & 0 & \mu & 0\\ 0 & 0 & 0 & 0 & 0 & \mu\end{bmatrix}\begin{bmatrix}\epsilon_{11}\\ \epsilon_{22}\\ \epsilon_{33}\\ 2\epsilon_{23}\\ 2\epsilon_{13}\\ 2\epsilon_{12}\end{bmatrix} \tag{11}$$

where $\mu$ and $\lambda$ are defined as (12) and (13), respectively:

$$\mu = G = \frac{E}{2(1+v)} \tag{12}$$

$$\lambda = \frac{vE}{(1+v)(1-2v)} \tag{13}$$

The inverse of this relation, on the other hand, can be thus, expressed as (14):

$$\epsilon = S\sigma \tag{14}$$

where $S = D^{-1}$ with components:

$$\begin{bmatrix}\epsilon_{11}\\ \epsilon_{22}\\ \epsilon_{33}\\ 2\epsilon_{23}\\ 2\epsilon_{13}\\ 2\epsilon_{12}\end{bmatrix} = \frac{1}{E}\begin{bmatrix}1 & -v & -v & 0 & 0 & 0\\ -v & 1 & -v & 0 & 0 & 0\\ -v & -v & 1 & 0 & 0 & 0\\ 0 & 0 & 0 & 2(1+v) & 0 & 0\\ 0 & 0 & 0 & 0 & 2(1+v) & 0\\ 0 & 0 & 0 & 0 & 0 & 2(1+v)\end{bmatrix}\begin{bmatrix}\sigma_{11}\\ \sigma_{22}\\ \sigma_{33}\\ \sigma_{23}\\ \sigma_{13}\\ \sigma_{12}\end{bmatrix} \tag{15}$$

### 2.1.1 Plane Stress and Plane Strain

Now, many engineering problems can be assumed to be plane (2D), i.e. independent on one of the coordinates, when the stress vector $t$ normal to a plane of reference practically constant but also so small in comparison with the stress vectors on a such plane, so that $t = \sigma \cdot \hat{n} \approx 0$. Examples of such plane problems are found, for instance, in those structures highly rigid or thick in the out-of-plane direction, such as tunnels **Fig. 4 Left**, Gravity Retaining walls **Fig. 4 Center** or in soil mechanics **Fig. 4 Right**.

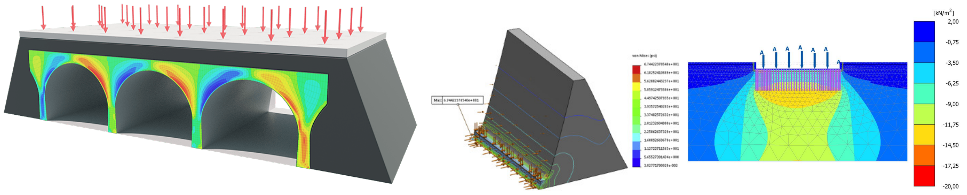

**Figure 4.** Examples of engineering problems where plane stress and plane strain assumptions are valid. (Left) Underground tunnel, (Center) Concrete Retaining Wall, (Right) Soil mechanics.

For such cases, the stress tensor is reduced to only three components as shown in **Fig. 5** that satisfy the linear elasticity equation (16):

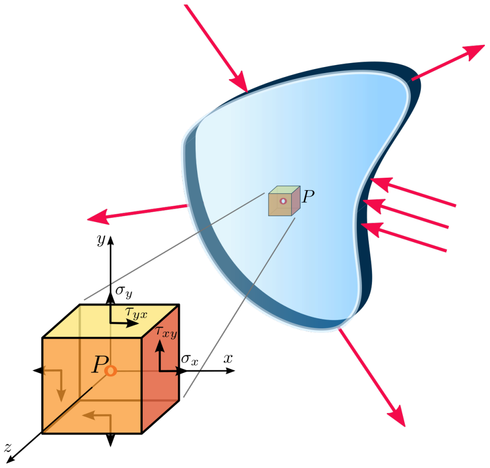

**Figure 5.** Plane Stress components.

$$\begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix} = \frac{E}{1-v^2} \begin{bmatrix} 1 & v & 0 \\ v & 1 & 0 \\ 0 & 0 & \frac{1-v}{2} \end{bmatrix} \begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{xy} \end{bmatrix} \tag{16}$$

or its inverse relation (17):

$$\begin{bmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{xy} \end{bmatrix} = \frac{1}{E} \begin{bmatrix} 1 & -v & 0 \\ -v & 1 & 0 \\ 0 & 0 & (1+v) \end{bmatrix} \begin{bmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{xy} \end{bmatrix} \tag{17}$$

## 2.2 Finite Element formulation

Hooke's law equation (10) can written in in terms of displacement gradient as (18) for any continuum:

$$\sigma = -D\epsilon = -D\nabla u \tag{18}$$

Now, by applying the force balance principle for each of the three main components in plane stress from **Fig. 5** the following equations (19) and (20) are obtained, where $bx$ and $b_y$ are the body force components over the whole continuum domain:

$$\frac{\partial \sigma_{xx}}{\partial x} + \frac{\partial \sigma_{xy}}{\partial y} + b_x = 0 \tag{19}$$

$$\frac{\partial \sigma_{yx}}{\partial x} + \frac{\partial \sigma_{yy}}{\partial y} + b_y = 0 \tag{20}$$

From which then the general balance equation (21) can be derived:

$$\nabla v(x,y)^T \sigma + b = 0 \tag{21}$$

where $\nabla v(x,y)^T$ is defined in FE matrix notation as (22):

$$\nabla v(x,y)^T = \begin{bmatrix} \frac{\partial}{\partial x} & 0 & \frac{\partial}{\partial y} \\ 0 & \frac{\partial}{\partial y} & \frac{\partial}{\partial x} \end{bmatrix} \tag{22}$$

### 2.2.1 Weak form

Now, by integrating over the whole volume domain of the continuum (with a constant thickness $t$) and by considering the boundary conditions additional to the body force the weak form takes place as (23)

$$-\int_A \nabla v(x,y)^T \sigma t dA - \int_L \nabla v(x,y)^T \sigma_n t dL = \int_A v(x,y) b t dA \tag{23}$$

which can be rewritten in terms of the constitutive relation matrix as (24):

$$\int_A \nabla v(x,y)^T t D \nabla u dA - \int_L \nabla v(x,y)^T \sigma_n t dL = \int_A v(x,y) b t dA \tag{24}$$

### 2.2.2 Finite Element form

By applying the Galerkin's method to approximate the solution we make:

$$\nabla v(x,y)^T = (Bv(x,y))^T = v(x,y)^T B^T \tag{25}$$

and:

$$\nabla u = \nabla N a = Ba \tag{26}$$

where $N$ is the shape function matrix defined as (27), for a finite element of $n$ nodes:

$$N = \begin{bmatrix} N_1(x,y) & 0 & N_2(x,y) & 0 & \cdots & N_n(x,y) & 0 \\ 0 & N_1(x,y) & 0 & N_2(x,y) & \cdots & 0 & N_n(x,y) \end{bmatrix} \tag{27}$$

Thus, $B$ would be defined as (28):

$$B = \nabla N = \begin{bmatrix} \frac{\partial N_1(x,y)}{\partial x} & 0 & \frac{\partial N_2(x,y)}{\partial x} & 0 & \cdots & \frac{\partial N_n(x,y)}{\partial x} & 0 \\ 0 & \frac{\partial N_1(x,y)}{\partial y} & 0 & \frac{\partial N_2(x,y)}{\partial y} & \cdots & 0 & \frac{\partial N_n(x,y)}{\partial y} \\ \frac{\partial N_1(x,y)}{\partial y} & \frac{\partial N_1(x,y)}{\partial x} & \frac{\partial N_2(x,y)}{\partial y} & \frac{\partial N_2(x,y)}{\partial x} & \cdots & \frac{\partial N_n(x,y)}{\partial y} & \frac{\partial N_n(x,y)}{\partial x} \end{bmatrix} \tag{28}$$

with which now, the FE form equation can be formulated as (29):

$$v(x,y)^T \int_A B^T DBdA \cdot a = \int_A N^T btdA + \int_L N^T tdL \tag{29}$$

where $a$ represent the displacements to be found by solving such system in the form (30):

$$[K]\{a\} = \{F_b\} + \{F_l\} \tag{30}$$

where $K$ is the stiffness matrix (relationship between forces and displacements).

## 2.3 Interpolating Shape Functions

Now, there can be uncountable 2D finite element geometries with which to generate and solve the previous system of equations (30). Of the most common and simple ones are the 2D-four-node-billinear element **Fig. 6** and the three-node-triangular finite element **Fig. 7**.

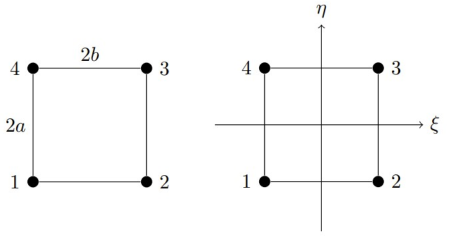

**Figure 6.** (Left) natural coordinate bilinear element, (Right) Parametric bilinear element.

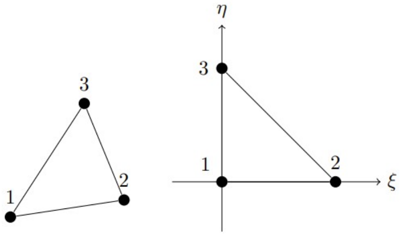

**Figure 7.** (Left) natural coordinate three-node triangular element, (Right) Parametric three-node triangular element.

### 2.3.1 Bilinear elements

The shape functions of the bilinear element in *natural coordinates* (**Fig. 6** (Left)) are defined as (31), and for parametric coordinates as (32). Parametric coordinated shape functions are preferred to work with as they allow the bilinear finite elements to have curved sides which would adapt better to complex and irregular surfaces. For such purpose, the Gaussian quadrature can be deployed by using two gauss points $n = 2$.

$$\begin{aligned}N_1 &= \tfrac{1}{4ab}(x-x_2)(y-y_4)\\ N_2 &= \tfrac{1}{4ab}(x-x_1)(y-y_3)\\ N_3 &= \tfrac{1}{4ab}(x-x_4)(y-y_2)\\ N_4 &= \tfrac{1}{4ab}(x-x_3)(y-y_1)\end{aligned} \tag{31}$$

$$\begin{aligned}\bar{N_1} &= \tfrac{1}{4}(\xi-1)(\eta-1)\\ \bar{N_2} &= -\tfrac{1}{4}(\xi+1)(\eta-1)\\ \bar{N_3} &= \tfrac{1}{4}(\xi+1)(\eta-1)\\ \bar{N_4} &= -\tfrac{1}{4}(\xi-1)(\eta+1)\end{aligned} \tag{32}$$

Similar as for the case in 2D, the quality of the results will depend directly on these shape functions. There are other bilinear elements that consist of more nodes (until 20 or more nodes) from which better interpolating results can be obtained, although with more computational power. In **Fig. 8** it is shown the interpolation shape of $N_1$ and $N_2$ for a four-node bilinear element:

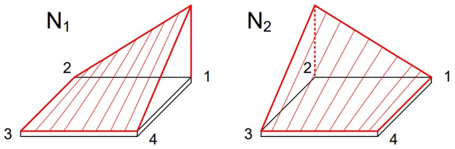

**Figure 8.** Interpolation functions for a 4-node bilinear element.

### 2.3.2 Linear Triangle element

For these elements the following interpolation shape functions apply, both in natural coordinates (33) and in parametric coordinates (34):

$$\begin{aligned} N_1 &= \tfrac{1}{2A}(x_2y_3 - x_3y_2 + (y_2 - y_3)x + (x_3 - x_2)y) \\ N_2 &= \tfrac{1}{2A}(x_3y_1 - x_1y_3 + (y_3 - y_1)x + (x_1 - x_3)y) \\ N_3 &= \tfrac{1}{2A}(x_1y_2 - x_2y_1 + (y_1 - y_2)x + (x_2 - x_1)y) \end{aligned} \tag{33}$$

$$\begin{aligned} \bar{N_1} &= 1 - \zeta - \eta \\ \bar{N_2} &= \zeta \\ \bar{N_3} &= \eta \end{aligned} \tag{34}$$

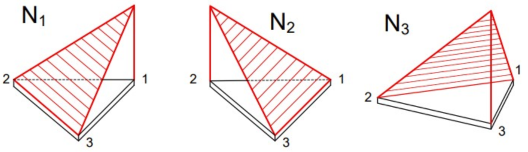

**Figure 9.** Interpolation functions for a 3-node triangular element.

# 3 Finite Element Analysis

## 3.1 Matrix formulation - Functional interpolation

Thus, when using these finite elements in parametric coordinates through the Gaussian Quadrature, the stiffness matrix $[K]$ of equation (30) would be computed as (35):

$$[K] = \int_{-1}^{1}\int_{-1}^{1} B(\zeta,\eta)^T D B(\zeta,\eta) det[J] d\zeta d\eta \tag{35}$$

where $[J]$ is *Jacobian* matrix defined as (36) which makes it possible to pass from one system of coordinates to another:

$$J = \begin{bmatrix} \frac{\partial N(\zeta,\eta)}{\partial\zeta} & \frac{\partial N(\zeta,\eta)}{\partial\eta} \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \frac{\partial x}{\partial\zeta} & \frac{\partial x}{\partial\eta} \\ \frac{\partial y}{\partial\zeta} & \frac{\partial y}{\partial\eta} \end{bmatrix} \tag{36}$$

Whereas the body force vector would be computed as (37):

$$\{F_b\} = \int_{-1}^{1}\int_{-1}^{1} N^T P_{xy} t det[J] d\zeta d\eta \tag{37}$$

where $Pxy$ is the unit volume weight vector $P = [P_x, P_y]$, according the material used.

Finally, for the boundary force vector, the computation goes as following:

$$F_l = t\int_0^L [-\frac{1}{L}(z-L), \frac{z}{L}]^T Q_{xy} dz \tag{38}$$

where $Q_{xy}$ is the boundary pressure $[Q_x, Q_y]$.

### 3.1.1 Strains and Stresses

Once the nodal displacements $\{a\}$ from equation (30) have been found, then the strains $\epsilon$ can be computed by deploying equation (26) as (39) and the stresses $\sigma$ can be computed by deploying equation (18) as (40), respectively:

$$\{\epsilon\} = [B_{\zeta,\eta}]\{a\} \tag{39}$$

$$\{\sigma\} = [D]\{\epsilon\} \tag{40}$$

### 3.1.2 Compatibility of nodal stresses and strains

It is to note, however, that the computed stresses and strains from equation (40) and (39) were computed at the location of each Gauss Point in the finite element in question, thus, such values must be scaled to the real values that would have in the finite element's nodes (see **Fig. 10**) in natural coordinates. For this purpose, the following computations are carried out:

$$\sigma_{P_i} = N_i \cdot \sigma_i \quad i = 1, 2, ..., n - nodes \tag{41}$$

where $s$ and $r$ are computed as (42) and (43), respectively:

$$s_i = \frac{x_i}{\zeta_i} \tag{42}$$

$$r_i = \frac{y_i}{\eta_i} \tag{43}$$

and $N_i$ depends on the geometry of the element used:

**For bilinear elements:**

$$N_i = \begin{bmatrix} \frac{1}{4}(1+r_i)(1+s_i) & \frac{1}{4}(1+r_i)(1-s_i) & \frac{1}{4}(1-r_i)(1+s_i) & \frac{1}{4}(1-r_i)(1-s_i) \end{bmatrix} \tag{44}$$

**For linear triangular elements:**

$$N_i = \begin{bmatrix} 1 - r_i - s_i & s_i & r_i \end{bmatrix} \tag{45}$$

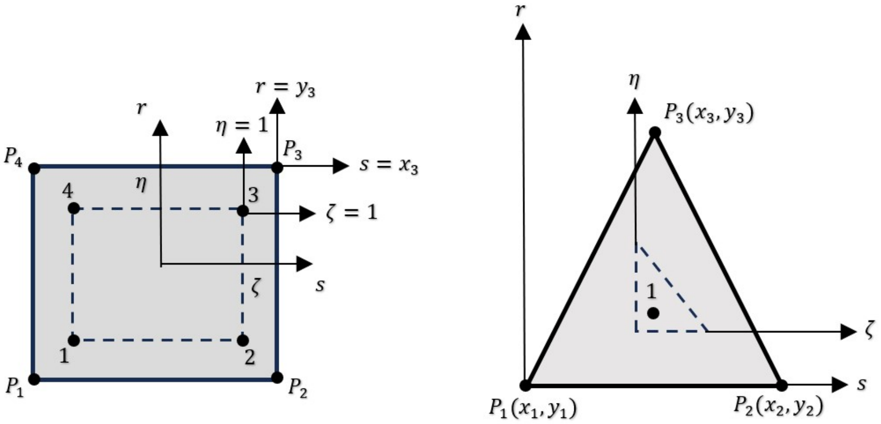

**Figure 10.** Scaling of stresses from a parametric element to a natural real coordinates.

Finally, once such stresses have been scaled to the real node coordinates, it has to be assured that every node of the mesh has a single stress or strain value. For this purpose an average of all values calculated for each node can be computed, or let the greatest calculated as the one considered for design purposes.

## 4 Illustrative example

Let us consider for example the following submarine concrete tunnel of **Fig. 11 (Left)**, which given its symmetry with respect to the vertical global axis its domain can be assumed as that of **Fig. 11 (Right)** so that a smaller mesh would be necessary to create reduce the computational demand.

The following geometry parameter values are considered for the analysis with reference of **Fig. 11 (Right)**:

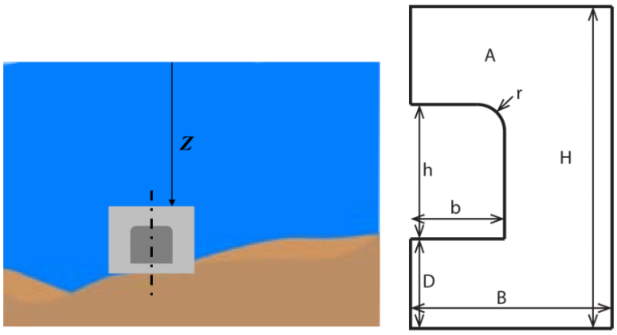

**Figure 11.** Submarine concrete tunnel considered for example for a FEA in plane stress and plane strain with bilinear and linear triangular elements.

$$
\begin{gathered}
H = 1000cm \\
B = 800cm \\
Z = 50000cm \\
D = 300cm \\
b = 400cm \\
h = 400cm \\
r = 40cm
\end{gathered}
$$

with a uniform out-of-plane thickness $t = 1cm$.

Regarding the material, a concrete of compressive strength equal to $f'_c = 300Kg/cm^2$ with a volume weight equal to $2400Kg/cm^3$ is used with the following mechanical properties:

$$
\begin{gathered}
E = 11000\sqrt{f'_c} \\
v = 0.2
\end{gathered}
$$

For this case, the hydrostatic lateral pressures are critical to consider for design and analysis. Thus, the following force diagram is taken for reference:

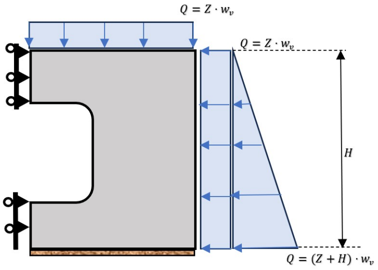

**Figure 12.** Force and pressure diagram for the concrete tunnel to be analysed.

## 4.1 FEA with bilinear elements

When using bilinear finite elements the following mesh of **Fig. 13** is created:

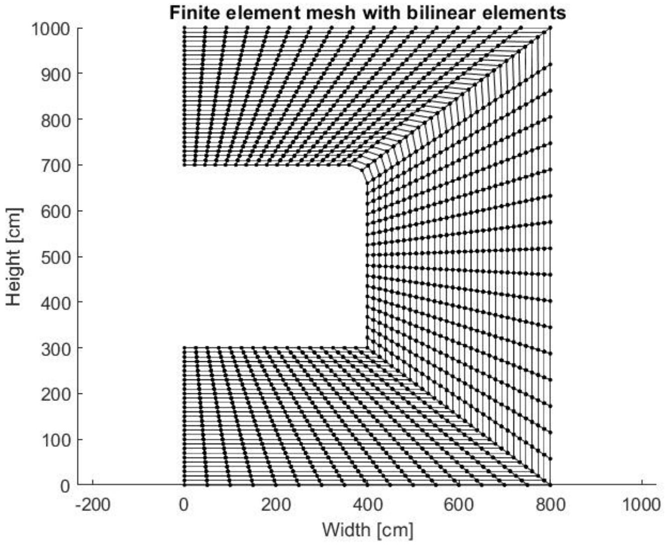

**Figure 13.** Finite Element mesh with bilinear elements for the submarine tunnel.

For analysis with the Gaussian Quadrature, 2 Gauss points are used on each axis direction so that the following results are obtained:

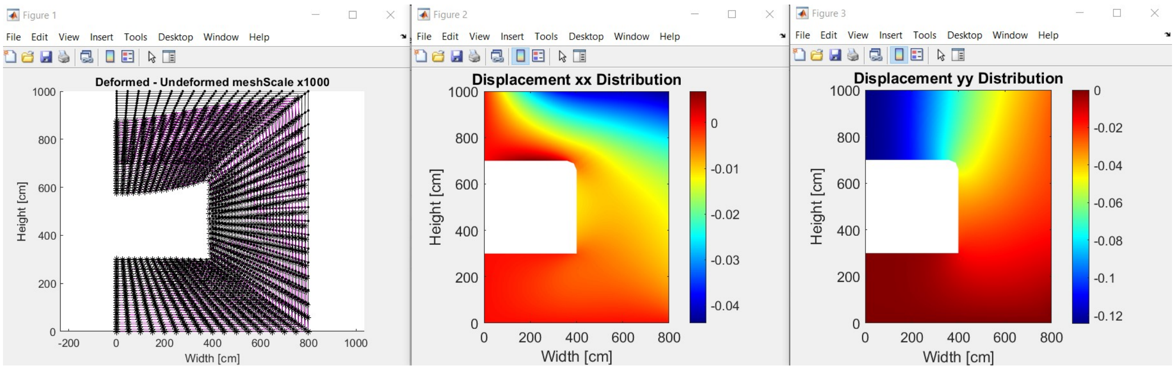

**Figure 14.** Results obtained for the FEA with bilinear elements. Top left: deformed and un-deformed mesh. Top center: displacements in the x direction. Top right: displacements in the y direction.

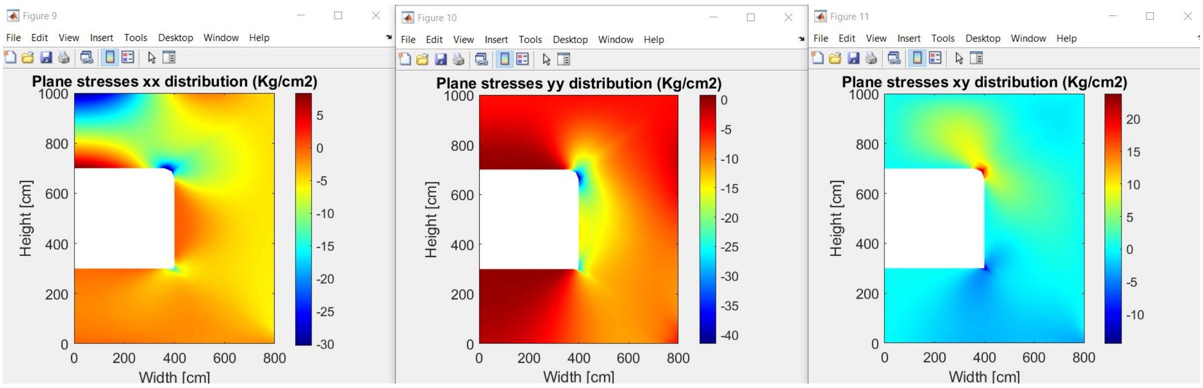
Figure 9
File Edit View Insert Tools Desktop Window Help
Plane stresses xx distribution (Kg/cm2)
1000
800
600
400
200
0
Height [cm]
0 200 400 600 800
Width [cm]
5
0
-5
-10
-15
-20
-25
-30
Figure 10
File Edit View Insert Tools Desktop Window Help
Plane stresses yy distribution (Kg/cm2)
1000
800
600
400
200
0
Height [cm]
0 200 400 600 800
Width [cm]
0
-5
-10
-15
-20
-25
-30
-35
-40
Figure 11
File Edit View Insert Tools Desktop Window Help
Plane stresses xy distribution (Kg/cm2)
1000
800
600
400
200
0
Height [cm]
0 200 400 600 800
Width [cm]
20
15
10
5
0
-5
-10

**Figure 15.** Results obtained for the FEA with bilinear elements. Top left: Stresses in the xx component. Top center: Stresses in the yy component. Top right: Stresses in the xy component.

## 4.2 FEA with linear triangular elements

On the other hand, when using linear triangular finite elements the following mesh of **Fig. 16** is generated:

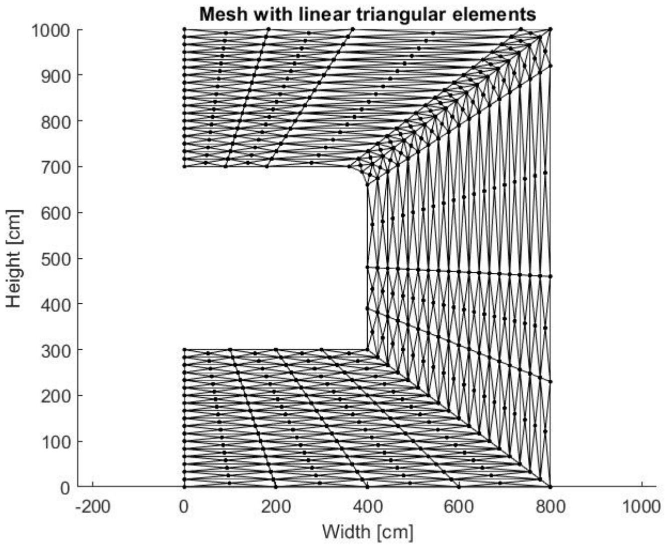

**Figure 16.** Finite Element mesh with linear triangular elements for the submarine tunnel.

For analysis with the Gaussian Quadrature, only 1 Gauss points is used so that the following results are obtained:

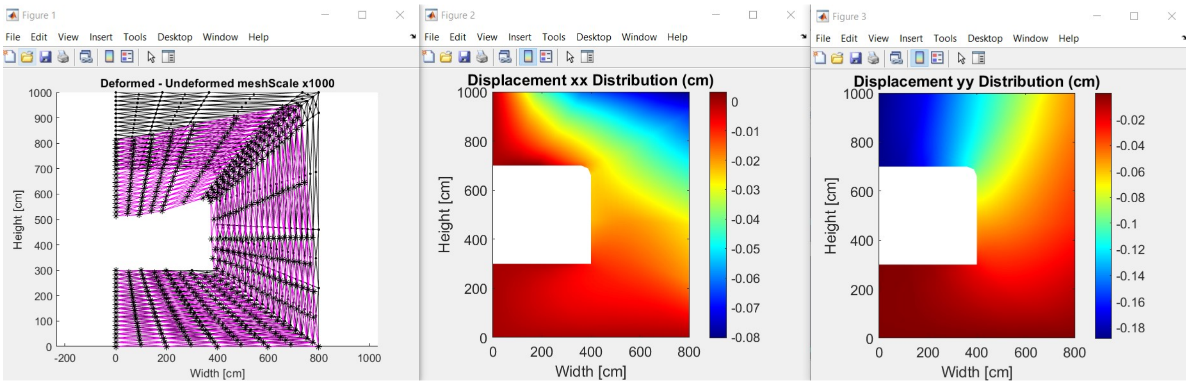

**Figure 17.** Results obtained for the FEA with linear triangular elements. Top left: deformed and un-deformed mesh. Top center: displacements in the x direction. Top right: displacements in the y direction.

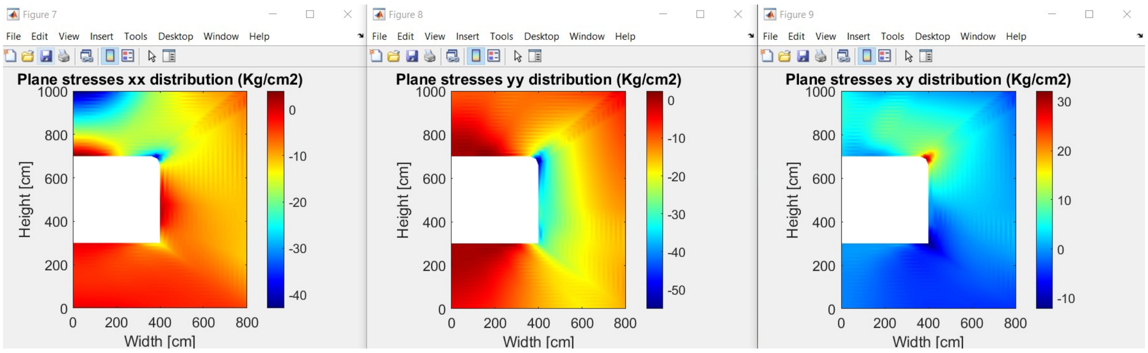

**Figure 18.** Results obtained for the FEA with linear triangular elements. Top left: Stresses in the xx component. Top center: Stresses in the yy component. Top right: Stresses in the xy component.

# References

[1] J.N. Reddy, An Introduction to the Finite Element Method, 2nd ed. McGraw-Hill, 1993

[2] R.I. Burden, D.J. Faires, A.M. Burden, Numerical Analysis, 10th ed. Cengage Learning, 2014