**PROYECTO FINAL**

Gabriel Castiblanco Cód. 3900168, Héctor Parra Cód. 3900181,

Ottorino Walton Páez Cód. 3900185

Universidad Militar Nueva Granada, Maestría en Ingeniería Mecatrónica

Profesor: PhD. Juan Manuel Lizarazo Marriega.

Implementación MEF para Placa con Concentrador de Esfuerzos Circular Centrado

Abstract- The finite element method allows the analysis of stresses and strains in objects; this paper seeks to create an algorithm able to mesh and find arrays employing this method, and compares the results with the analytical method and obtained specialized software.

**Resumen - El método de elementos finitos permite el análisis de esfuerzos y deformaciones en objetos, en este trabajo se busca crear un algoritmo capaz de enmallar y hallar las matrices que emplea este método, para luego comparar los resultados obtenidos con el método analítico y los obtenidos con un software especializado.**

*Palabras clave*— Placa Con Agujero Central, MEF, esfuerzos, deformaciones. ANSYS.

# INTRODUCCIÓN

Para utilizar el Método de Elementos Finitos, se debe tener en cuenta, que la solución es aproximada y depende de que tan fina sea la distribución de elementos seleccionada, sin exceder un límite de elementos que exija un gasto computacional alto, sin ofrecer una mayor precisión. Los usuarios de estas herramientas o métodos deben ser conscientes de estas limitaciones y tener en cuenta que una mayor cantidad de elementos no mejorara los resultados.

En este trabajo se plantean los siguientes objetivos fundamentales.

1. Desarrollo de pre procesó para determinar el enmallado a partir de la geometría.
2. Diseñar un algoritmo capaz de calcular esfuerzos y deformaciones a partir del reproceso.
3. Realizar la comparación entre los resultados obtenidos analíticamente, los obtenidos mediante un herramienta para MEF como ANSYS y los resultados mediante el algoritmo desarrollado.
4. Presentar de forma gráfica el resultado de los esfuerzos y deformaciones normalizadas.
5. Presentar en isolíneas los valores promedio de deformaciones y esfuerzos calculados.

# DESARROLLO

## **METODO ANALITICO**

Análisis de la placa por resistencia de materiales utilizando las curvas de concentradores de esfuerzo Figura 1.

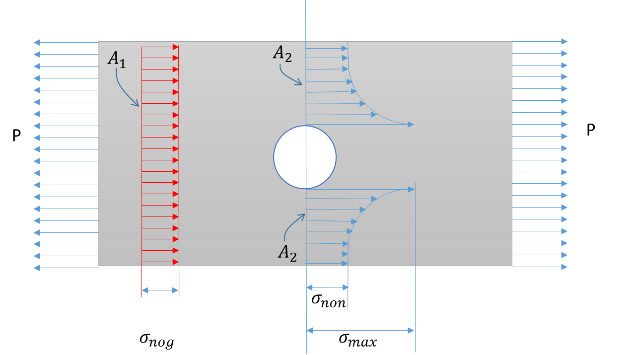


Figura 1Grafica deFactores de esfuerzos

# 

Deformación por módulo de elasticidad

Deformación , Young E

(Nominal grueso)

(Nominal neto)

Chapa perforada en el centro bajo condición de esfuerzo plano Figura 2.

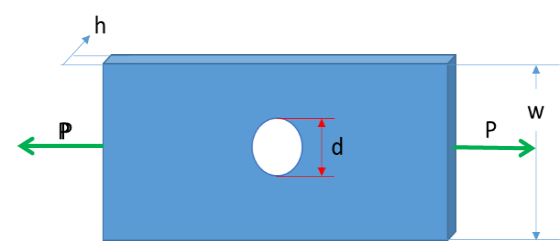


Figura 2 Chapa perforada

Para

+

Desarrollo para la fuerza de trabajo

Las características de la placa son: largo 30 pulgadas, altura 15 pulgadas, ancho 12 mm, agujero central diámetro 5 cm y tensión plana sometida a tracción constante en su extremo 5 Toneladas por metro, de acuerdo a la Figura 3.

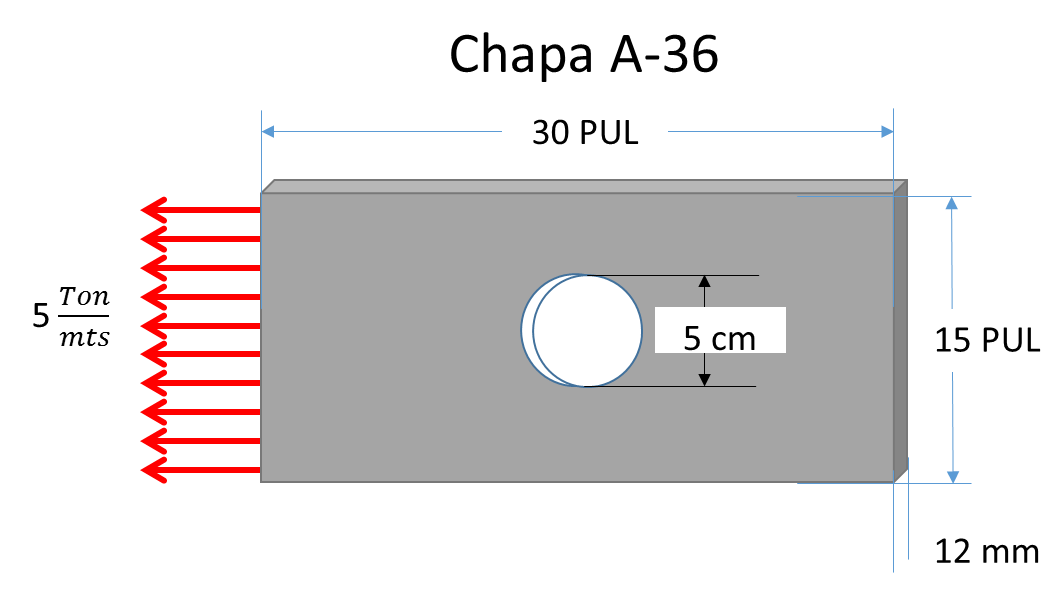


Figura 3 Placa con agujero central.

La carga P concentrada es

Relación

= 20

Material a analizar

Astm A36 estructural

KSI

Resultados de la simulación en ANSYS.

= 35.76 = 5187

Para 50.000 N

(4191 Lb carga P)= 1934.1 PSI

A continuación, se muestra las medidas en milímetros de un cuarto de placa esto con el objetivo de aprovechar la simetría de la pieza a analizar. Figura 4.

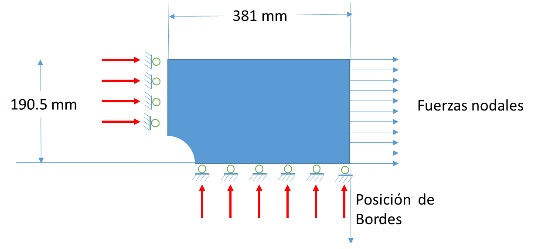


Figura 4 Placa con agujero central.

## **PRE PROCESO POR MEF PARA CREACION DE LA MALLA.**

La generación de la malla, es uno de los aspectos más importantes en elementos finitos. Es importante controlar el tamaño del elemento finito a utilizar, de modo que este sea adecuado para mantener acotado el error de la solución en los límites admisibles. Por otro lado, no se puede simplemente generar elementos muy pequeños en todo el modelo, ya que esto origina mallas, con muchos nodos, sobre todo en problemas complejos en tres dimensiones, lo cual implica mayor gasto computacional.

Para lograr el enmallado en forma automática y anti horaria se usó MatLab, la cual cuenta con una herramienta para el análisis de elementos finitos conocida como Pdetool, esta permite a partir de la geometría, crear el enmallado y realizar variaciones en la cantidad de nodos y elementos, a su vez permite exportar las matrices correspondientes a las coordenadas de cada nodo, la matriz de relaciones entre elementos matriz de conectividad, y matriz de entorno geométrico.

A continuación se muestran varios tamaños de enmallado posibles con esta herramienta.

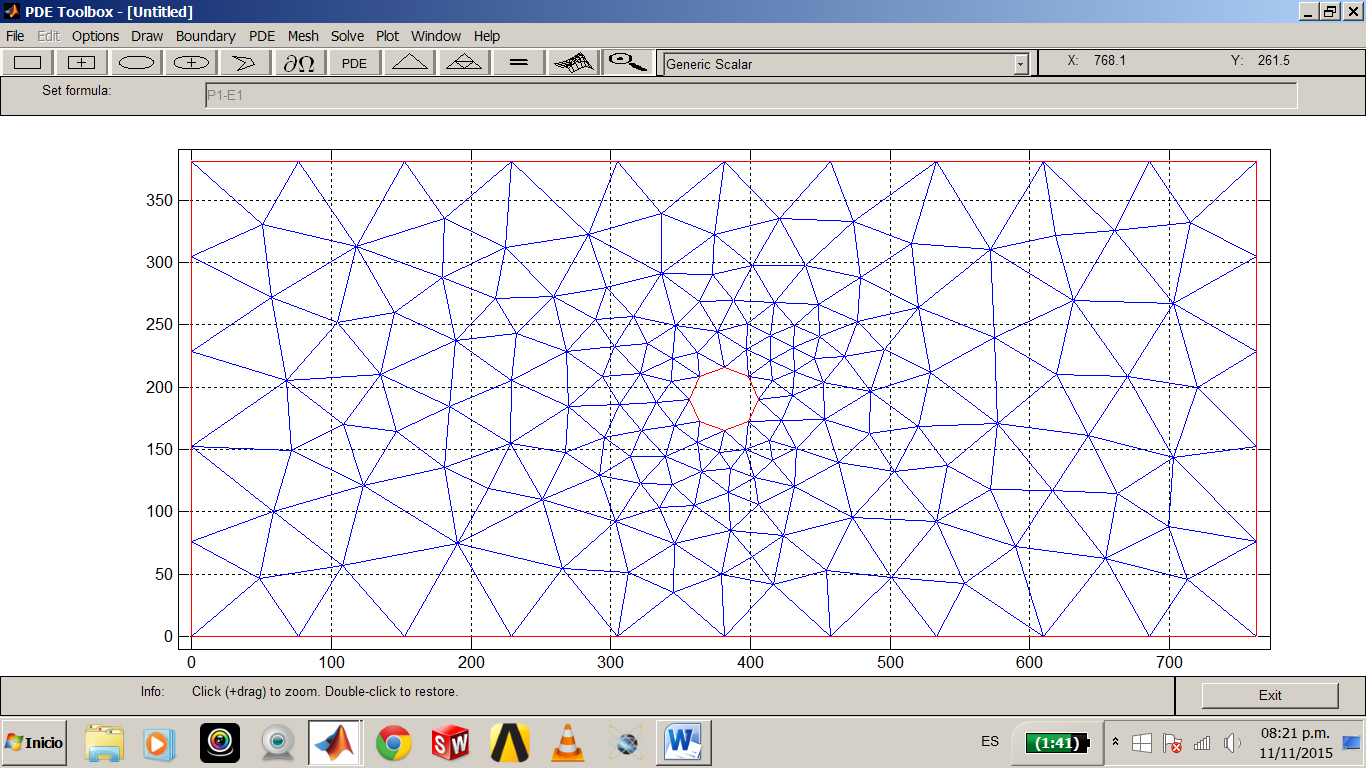
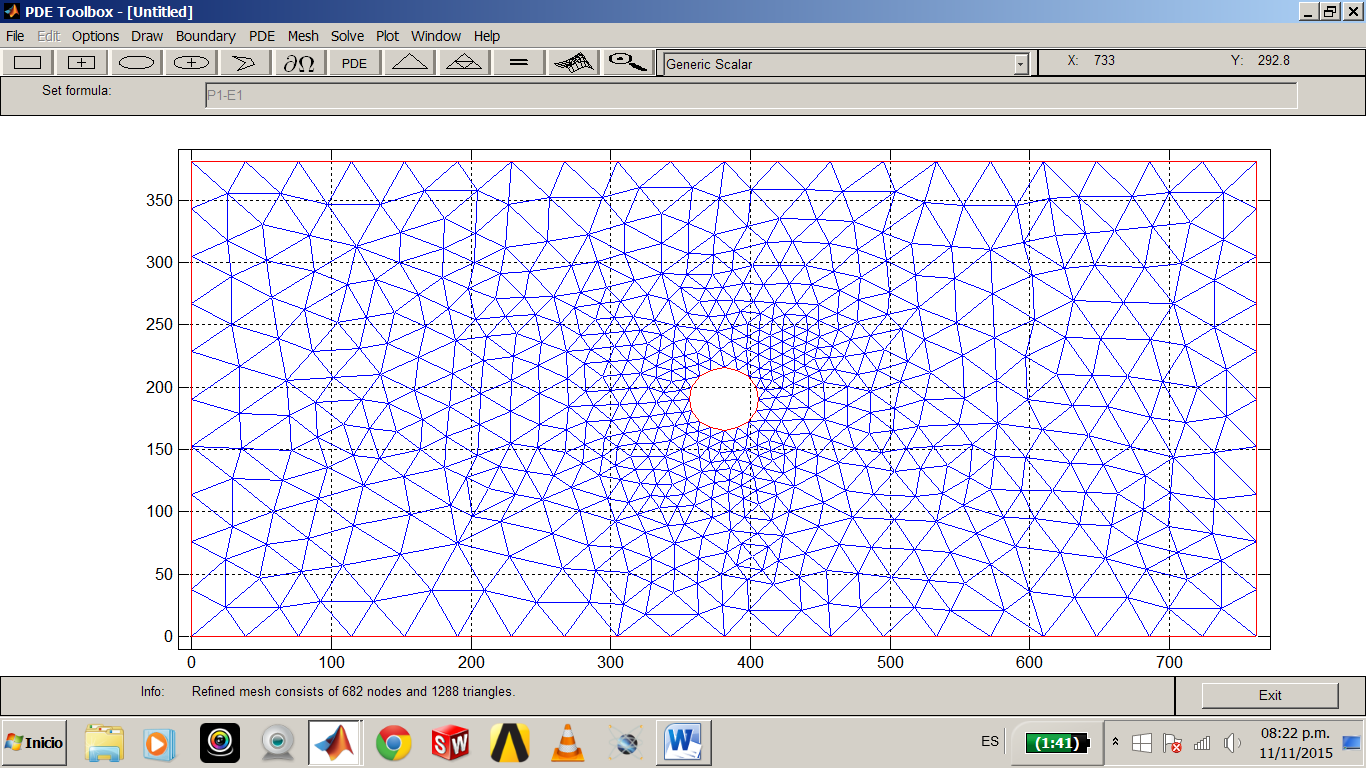


Figura 5 Malla inicial de 322 elementos.

Figura 6 Malla para 1288 triángulos y 682 nodos.

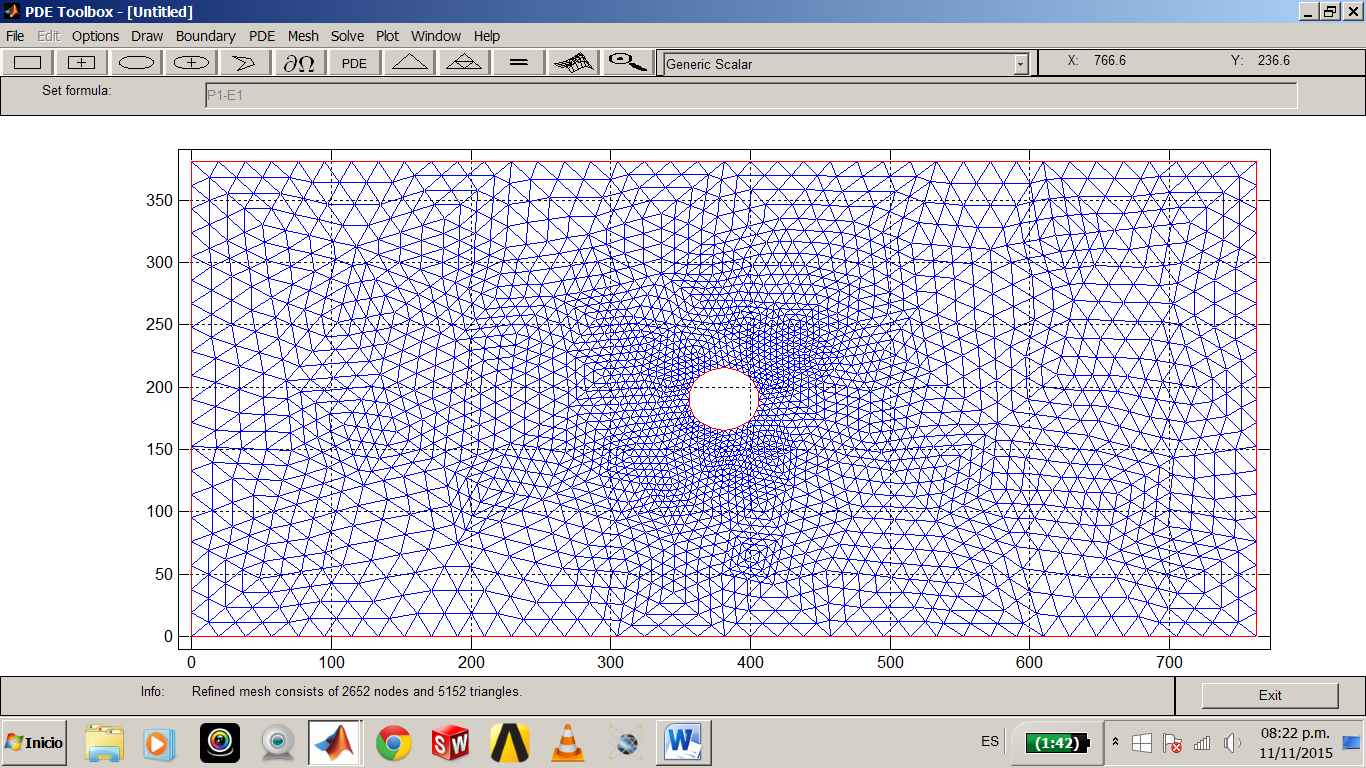


Figura 7 Malla para 5152 triángulos y 2652 nodos.

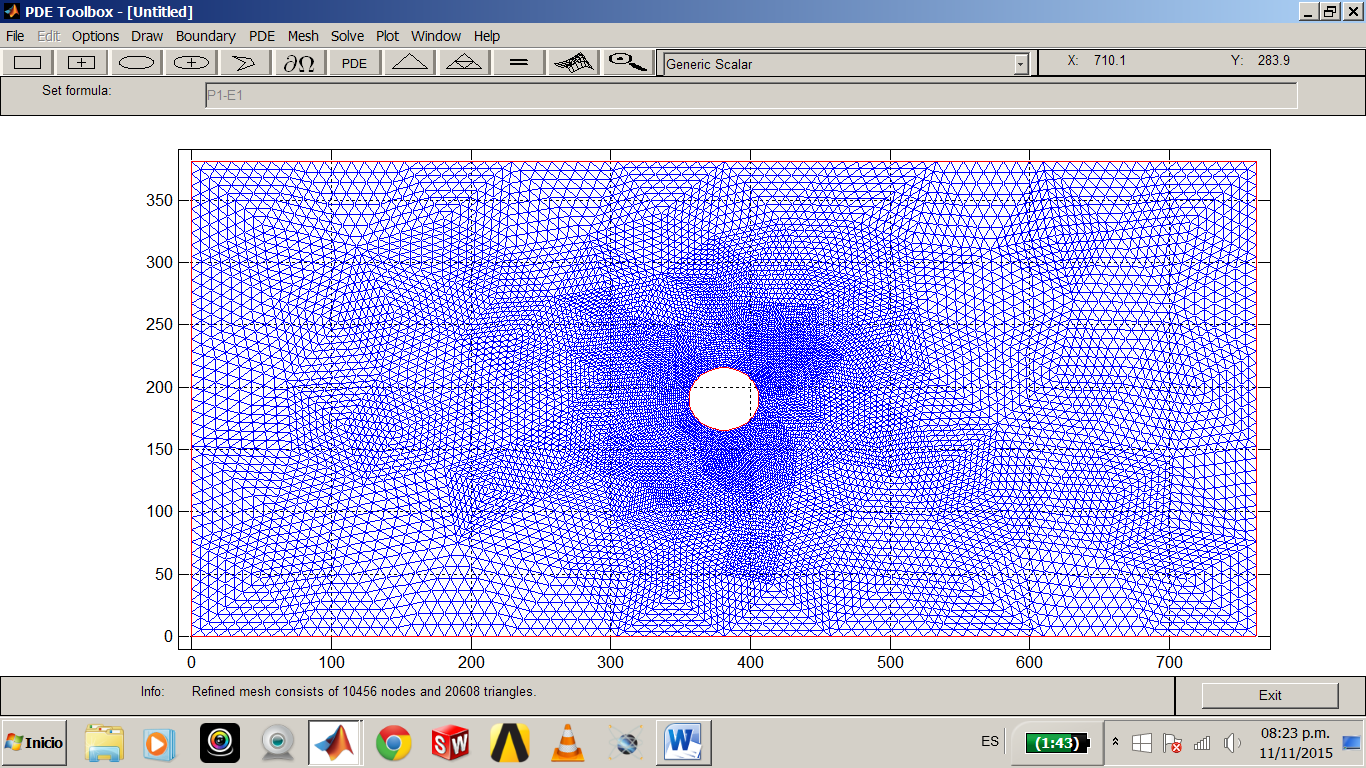


Figura 8 Malla para 20608 triángulos y 10456 nodos.

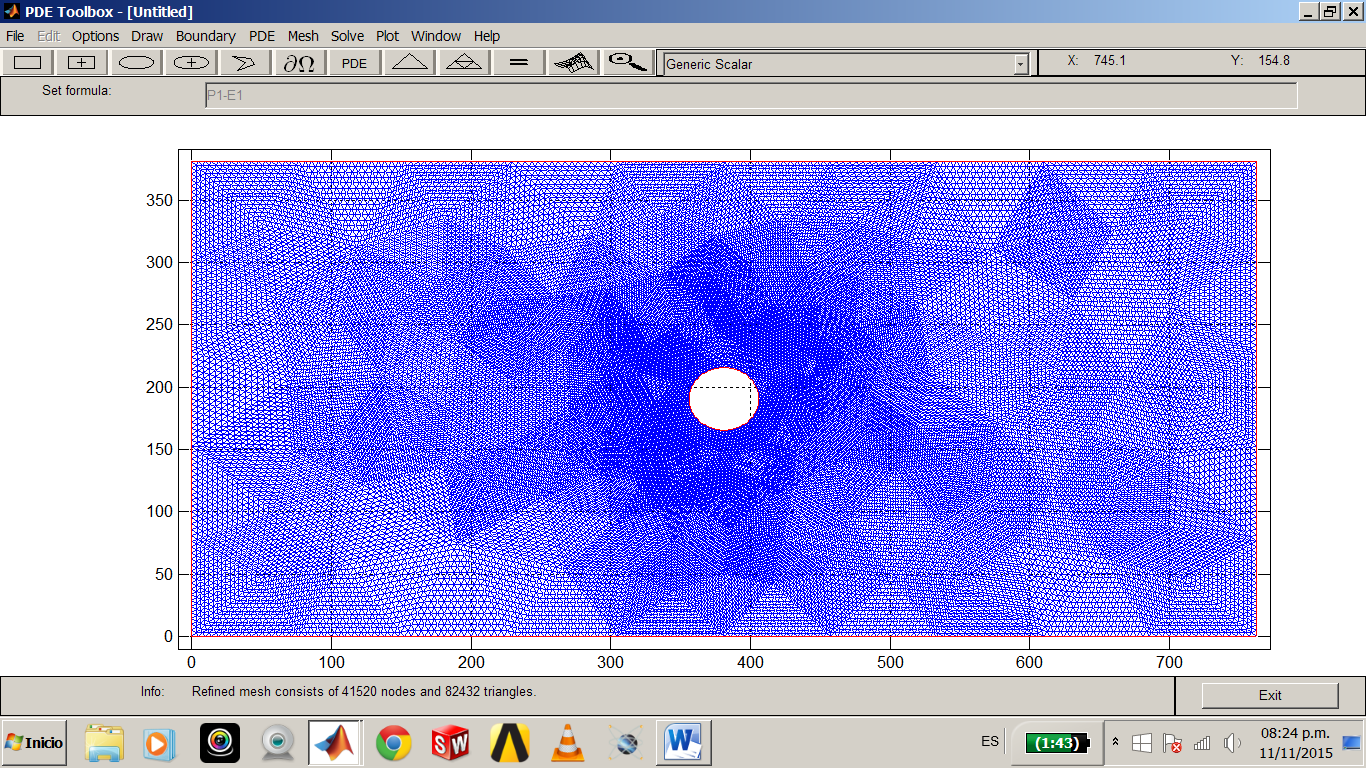


Figura 9 Malla para 82432 triángulos y 41520 nodos.

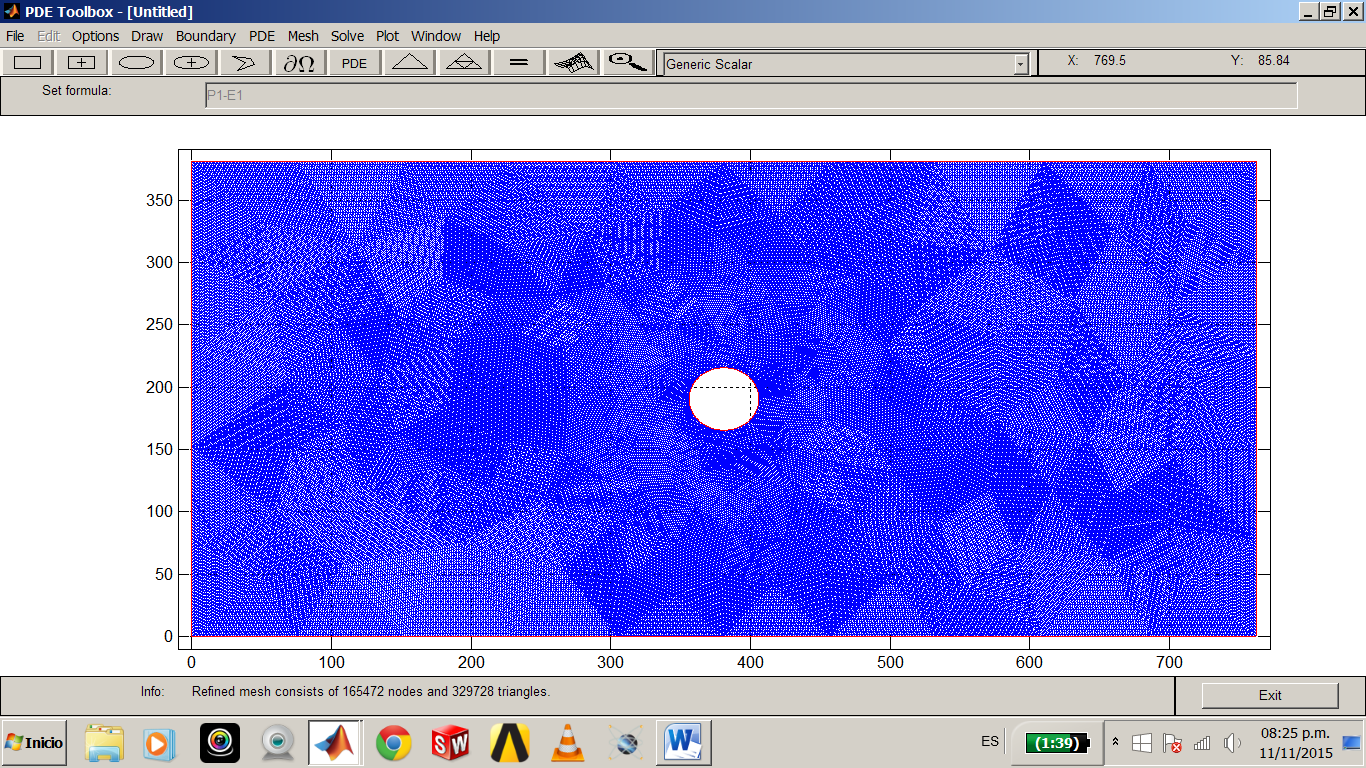
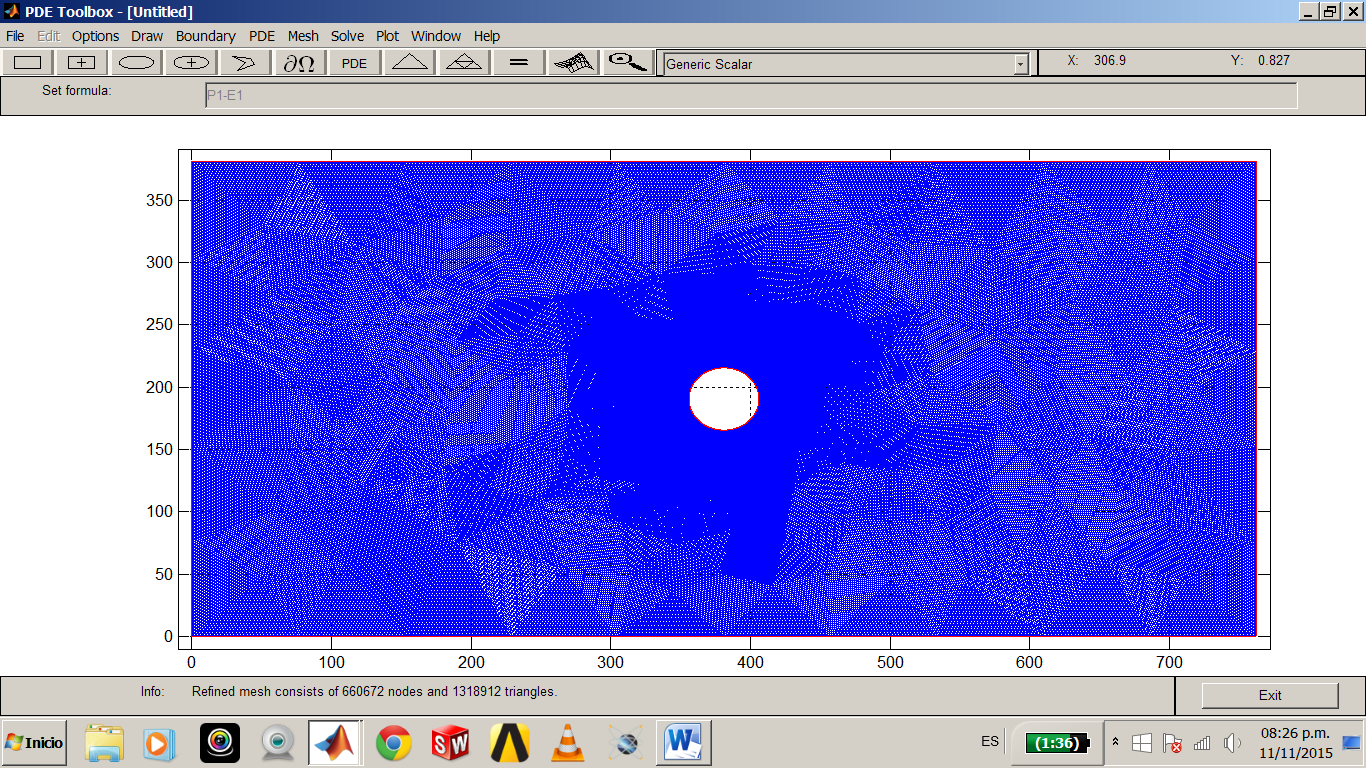


Figura 10 Malla para 329728 triángulos y 165472 nodos.

Figura 11 Malla para 1318912 triángulos y 660672 nodos.

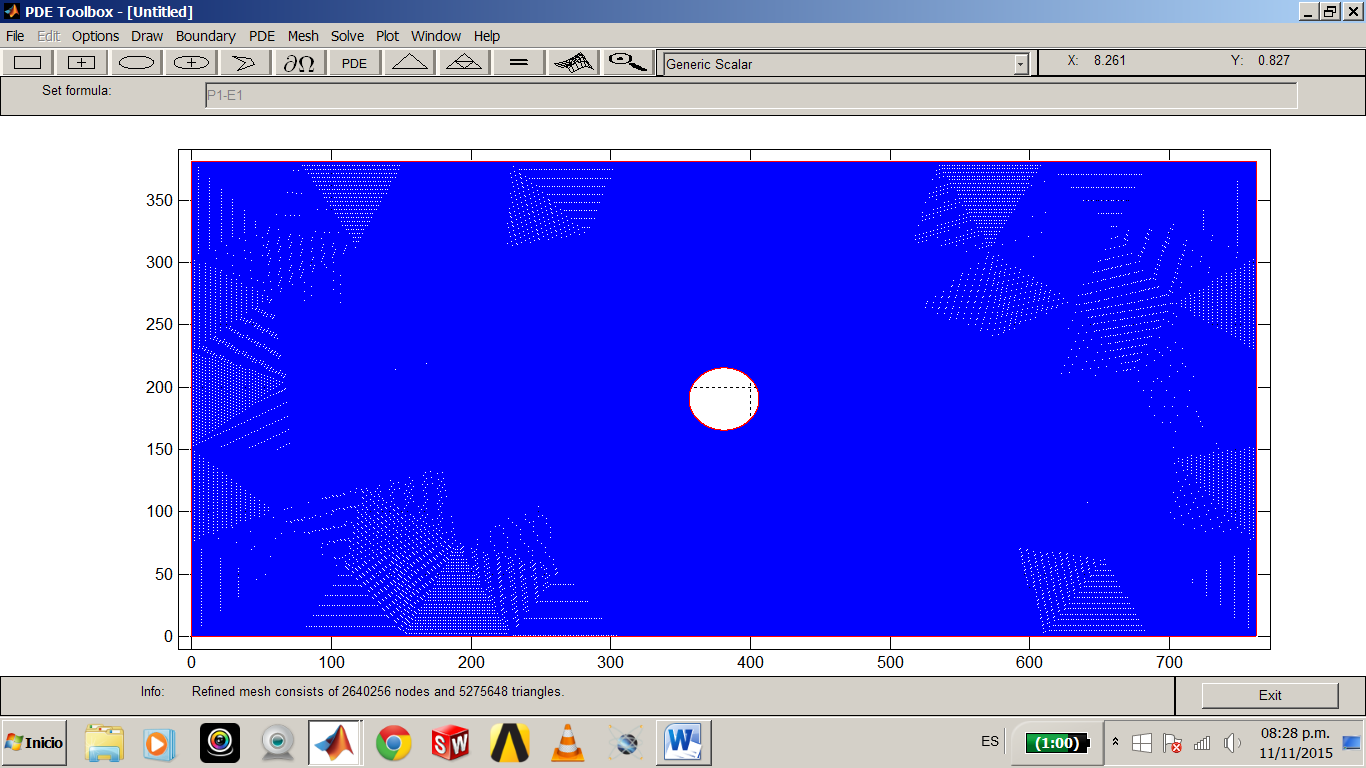


Figura 12 Malla para 5275648 triángulos y 2640256 nodos.

## **ALGORITMO PARA EL CALCULO DE MATRICES DE ELASTICIDAD.**

A continuación se realiza genera un algoritmo basándose en las matrices de elementos finitos vistas en clase.

Matriz de coordenadas [# nodos; (x y)]

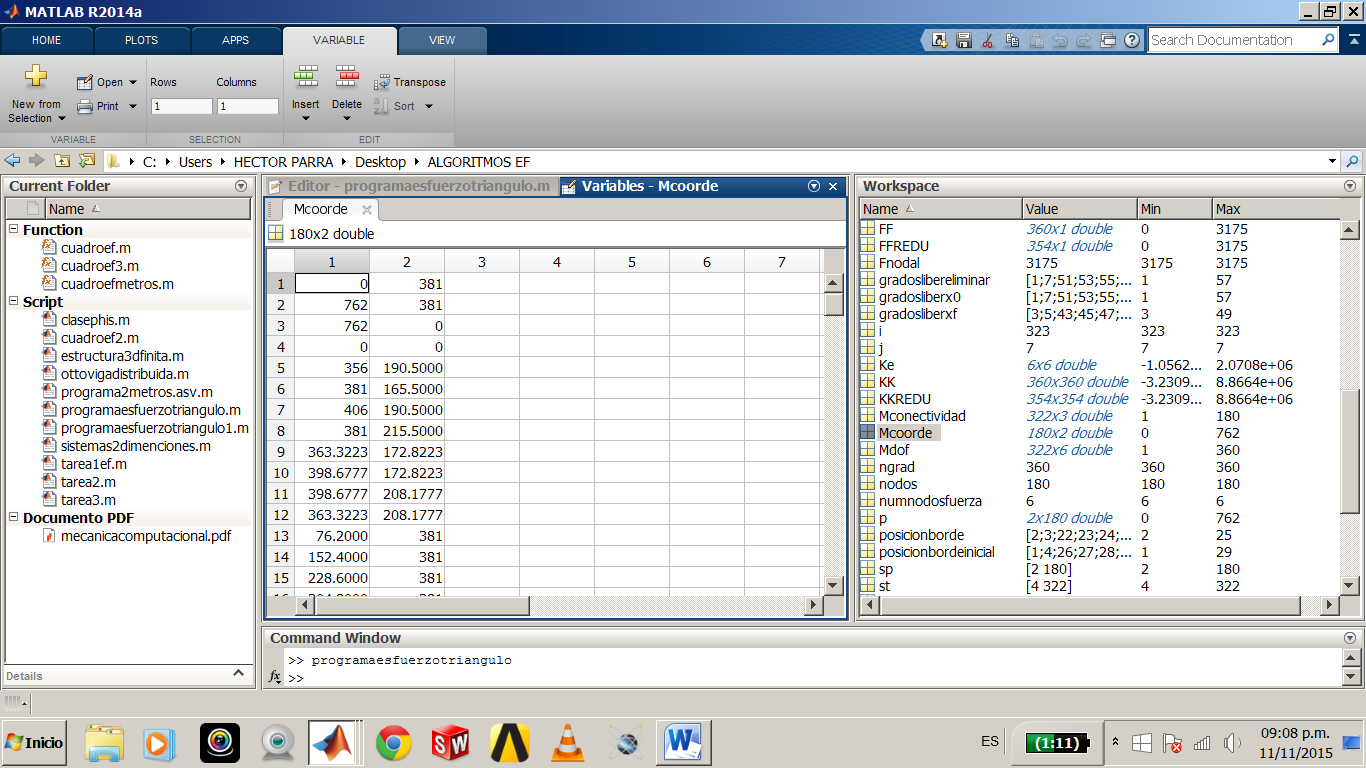


Figura 13 Matriz de coordenadas calculada

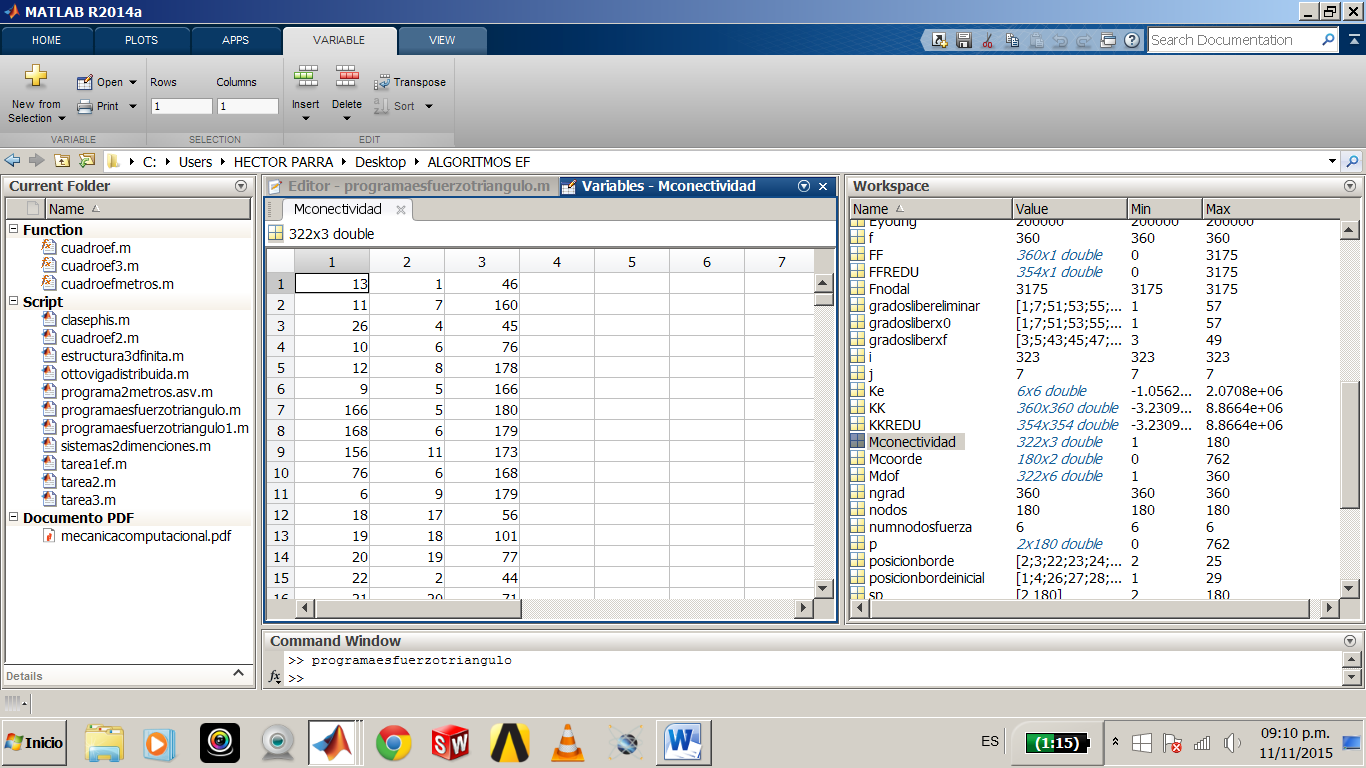
Matriz de conectividad [# elementos; 3 nodos (i j k)]

Matriz de deformación

Matriz de propiedades elásticas

Esfuerzo Plano

Deformación

Figura 14 Matriz de conectividad calculada.

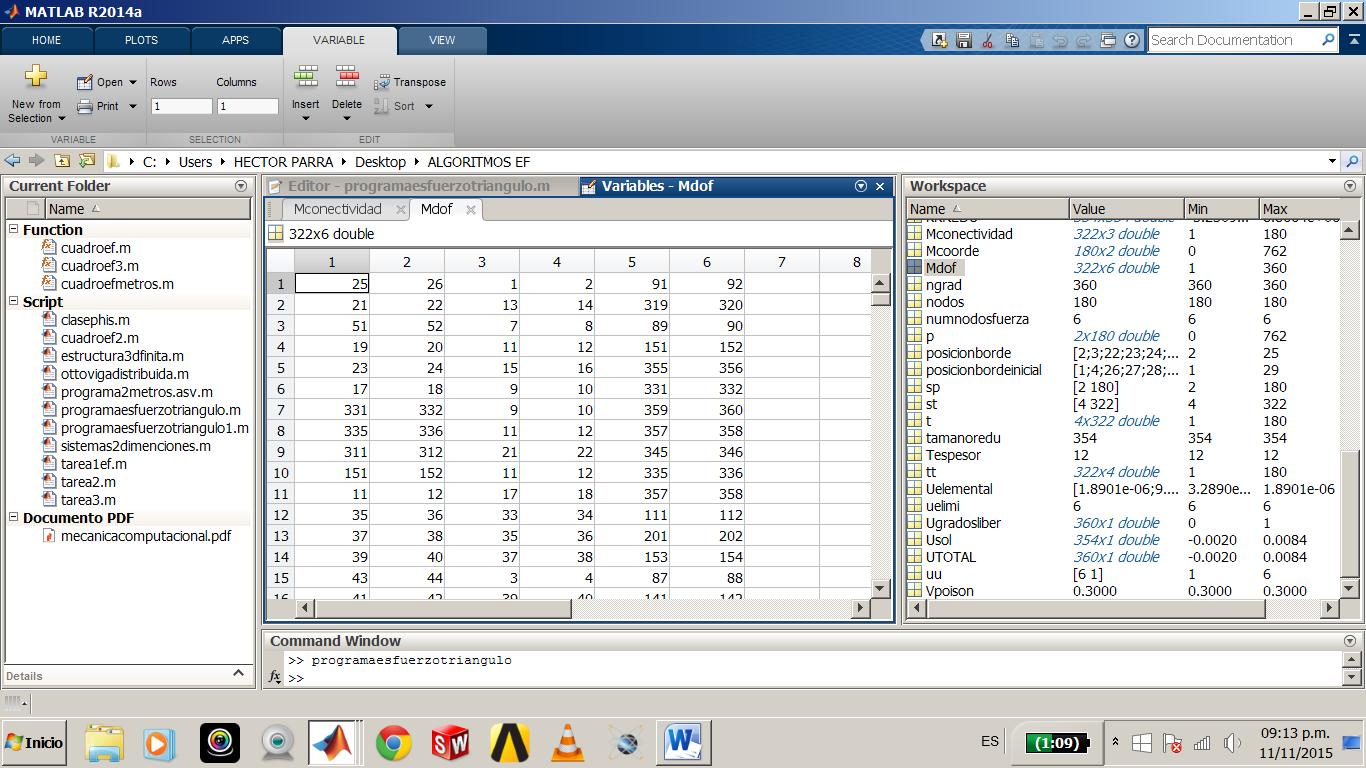


Figura 15 Matriz de grados de libertad (DOF).

# 

Figura 16 Matriz de elasticidad global generada.

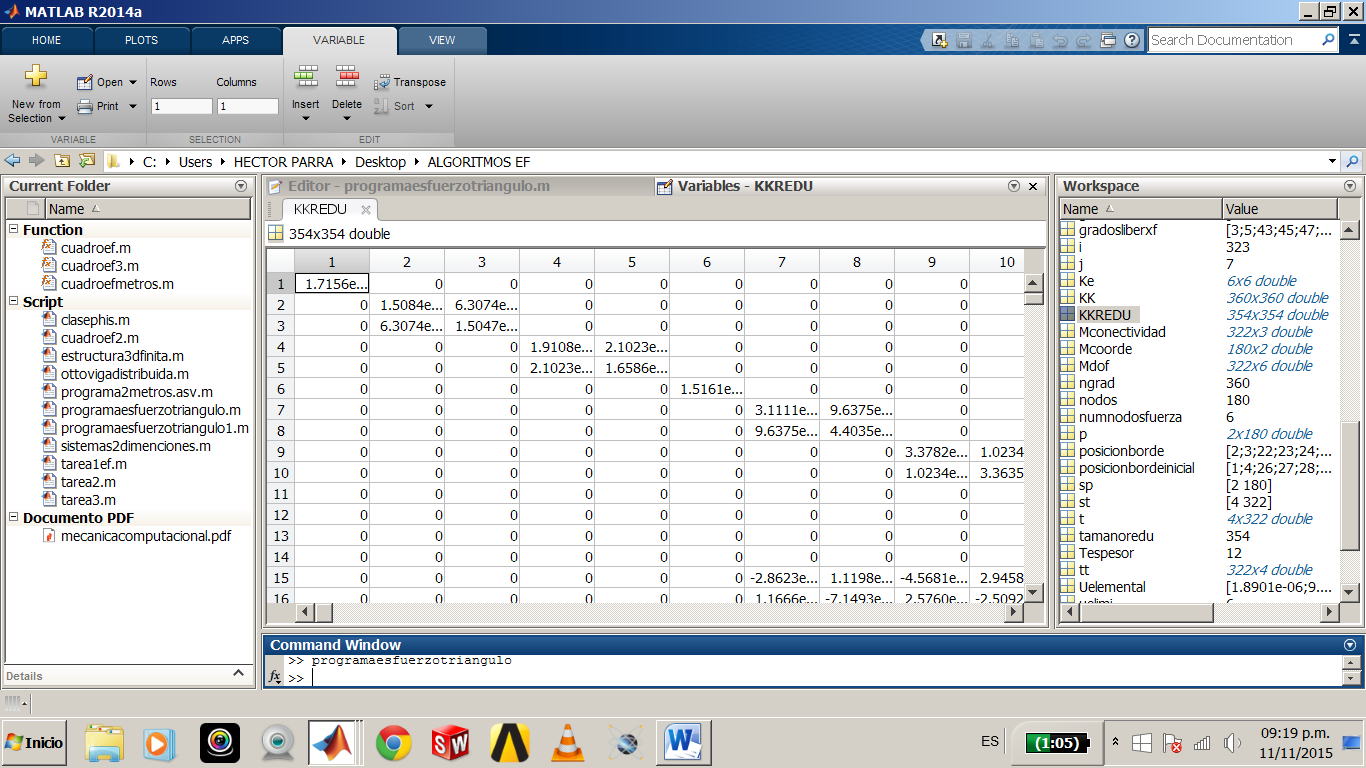


Figura 17 Matriz de elasticidad reducida generada.

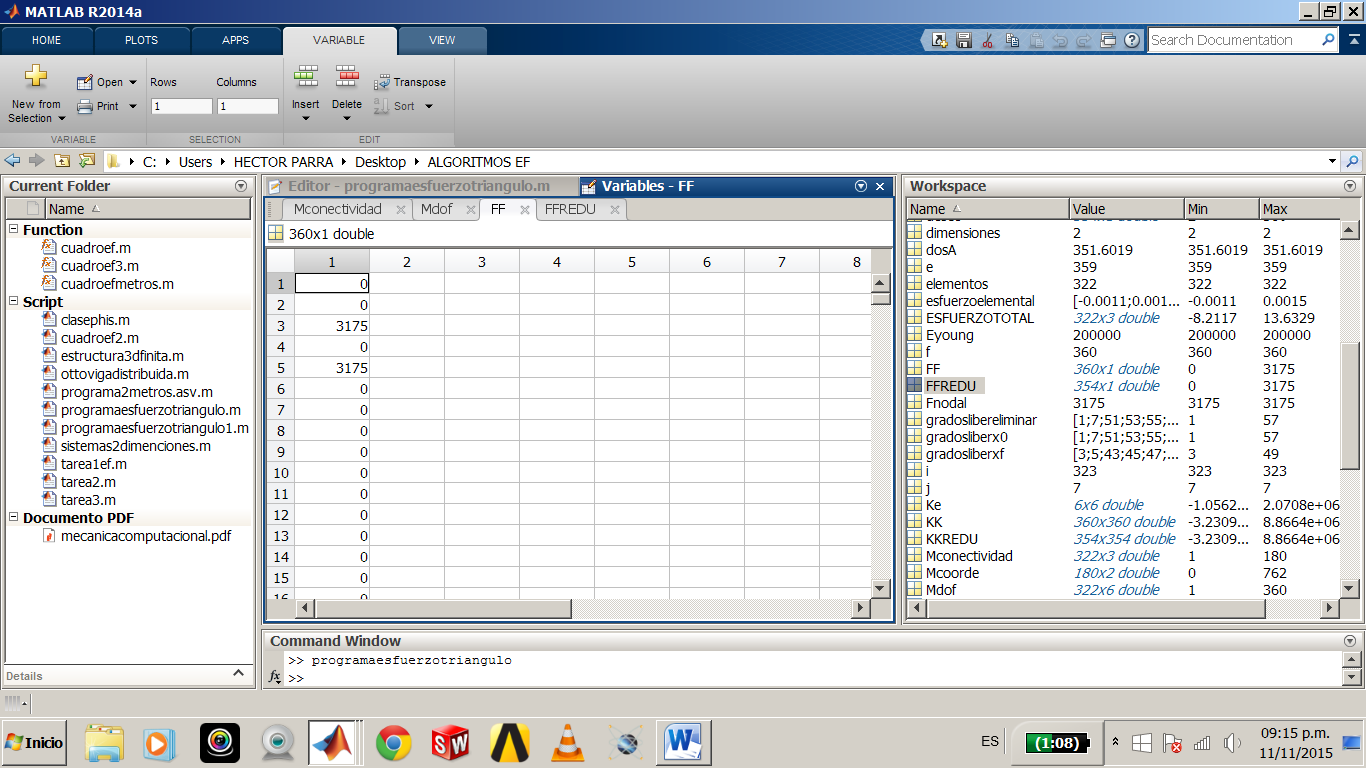


Figura 18 Matriz total fuerzas.

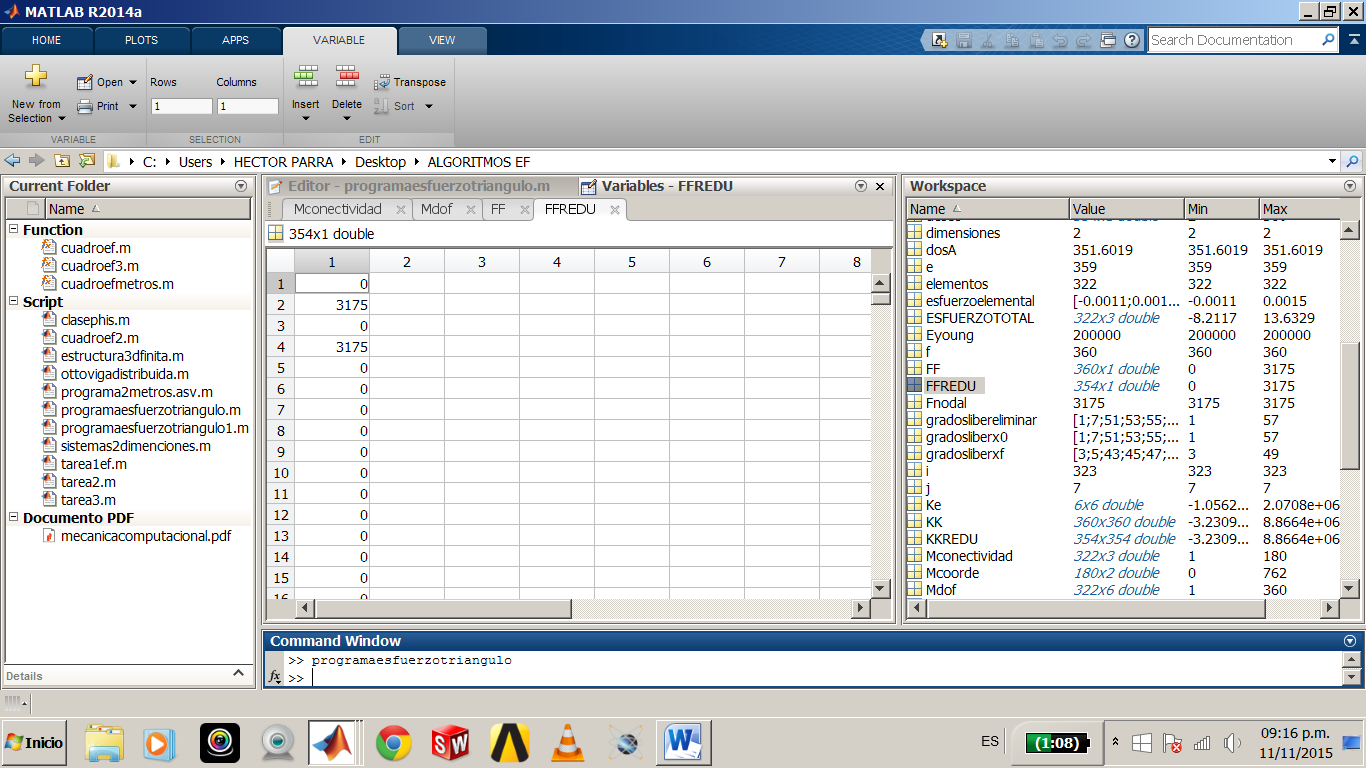


Figura 19 Matriz de fuerza reducida.

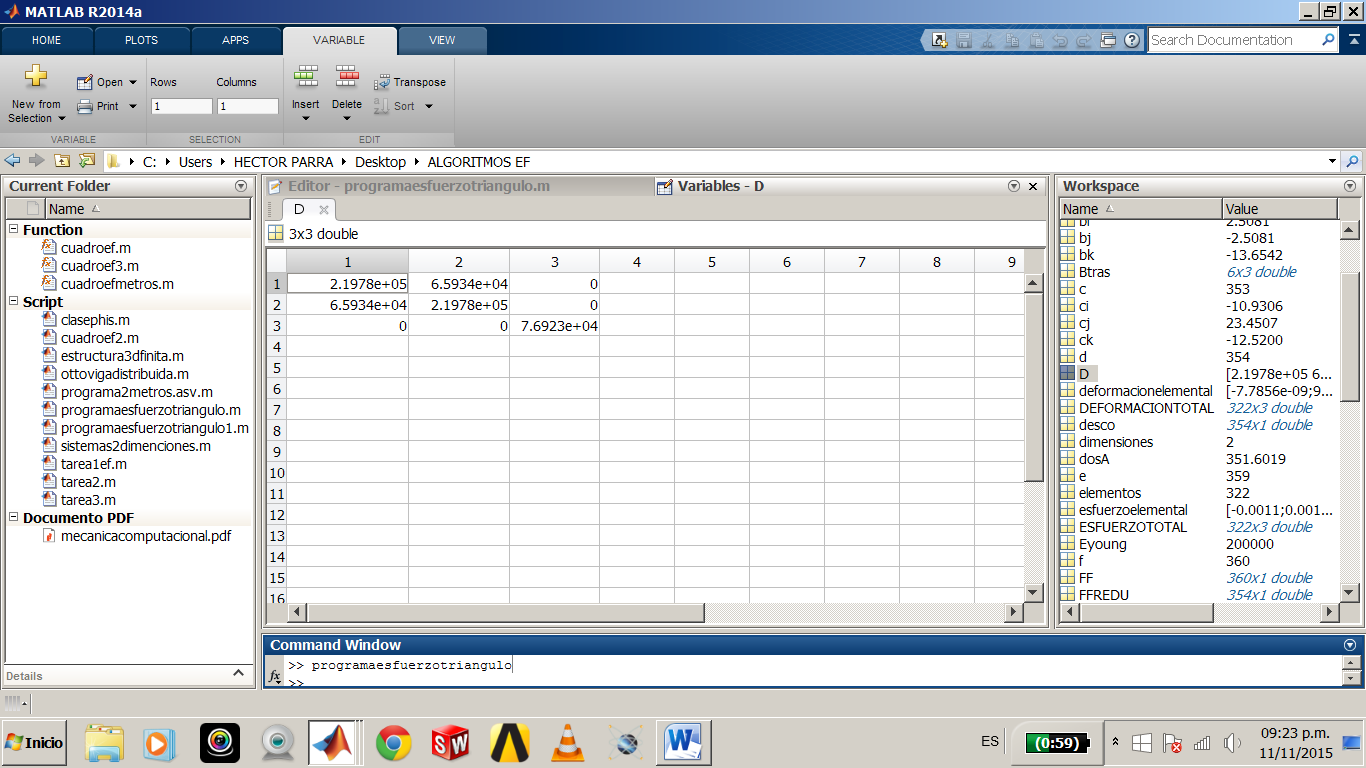


Figura 20 Matriz de propiedades elásticas.

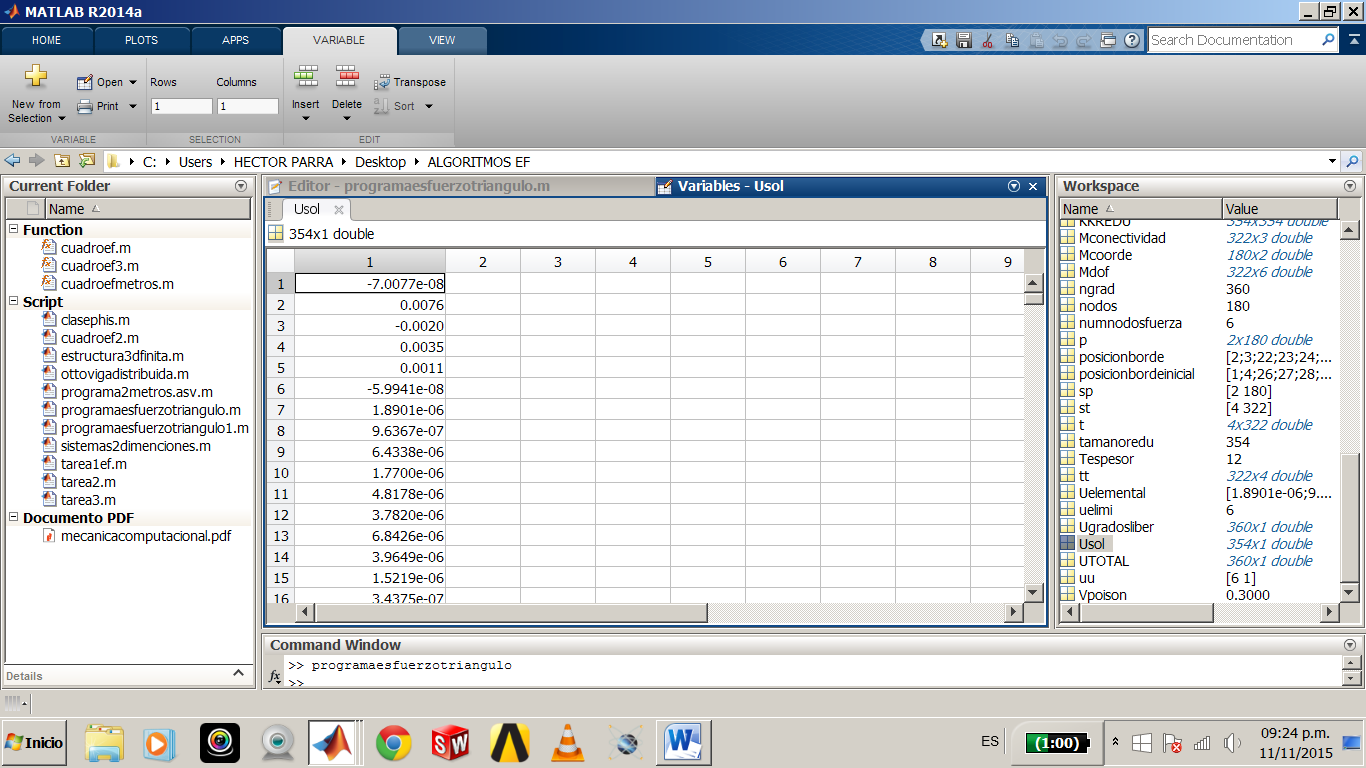


Figura 21. Matriz de desplazamiento por grados de libertad (Usol = KKREDU\FFREDU).

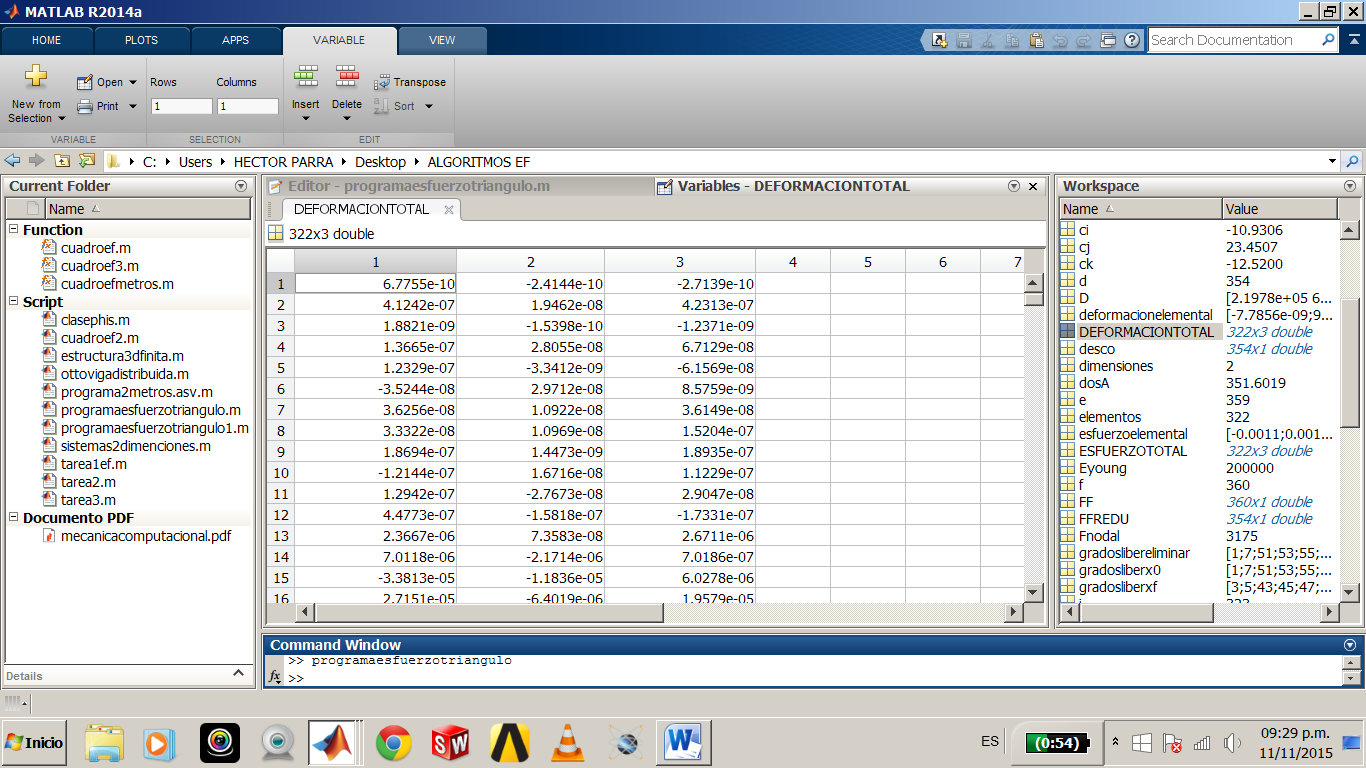
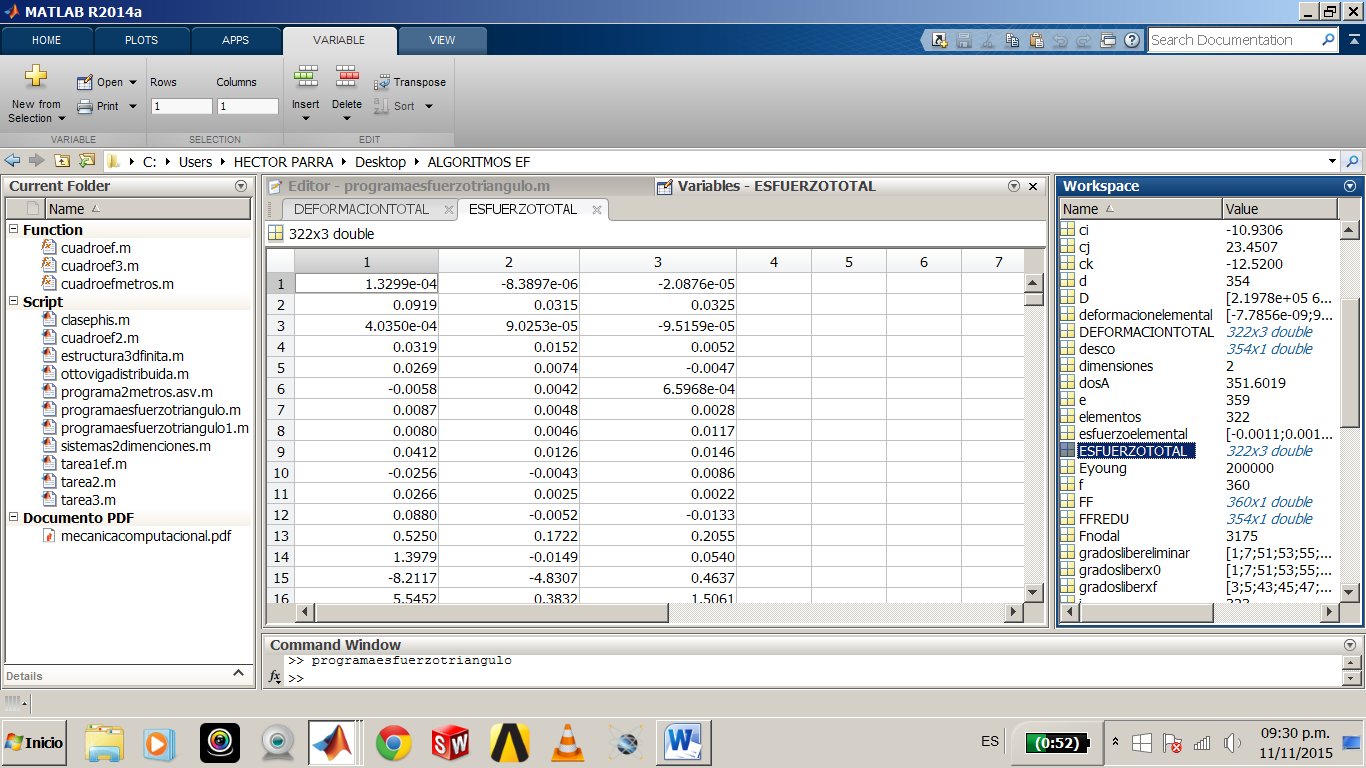


Figura 22 Matriz de deformación por elemento.

Figura 23. Matriz de esfuerzos por elemento.

## **DESARROLLO EN ANSYS**

### Simulación realizada con malla gruesa:

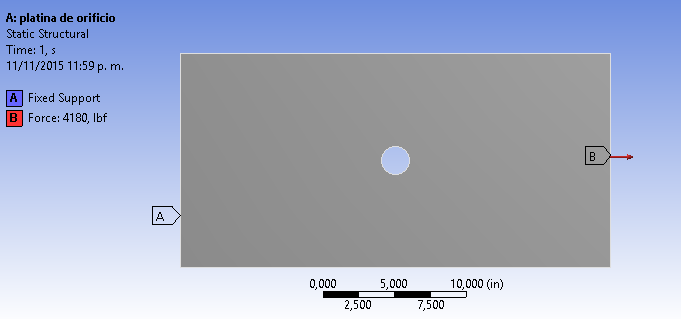


Figura 24 . Puntos de Soporte y Fuerza Aplicada.

Realizando el análisis de esfuerzo deformaciones en ANSYS se obtiene los siguientes resultados.

Los cálculos fueron realizados mediante la teoría de esfuerzos equivalentes de Von Mises. que corresponde a uno de los métodos más utilizados en ANSYS para el cálculo de esfuerzos y deformaciones en elementos con cargas estáticas.Figura 25

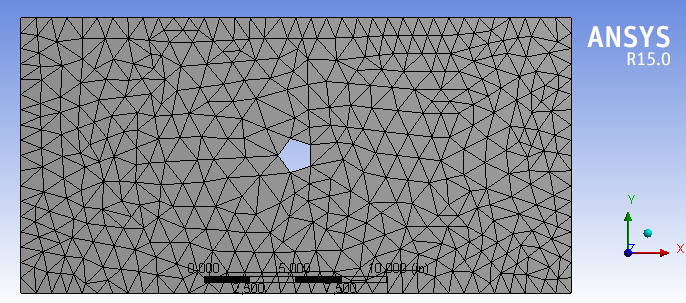


Figura 25. Distribución de malla de 985 Nodos y 2674 elementos triangulares de tres nodos.

Como puede observarse para una malla gruesa, se presenta un fenómeno en el concentrador de esfuerzos circular, que no permite con los triángulos de la malla formar de manera adecuada la circunferencia.

Como se observa en la gráfica de la Figura 25, los triángulos forman un pentágono que ocasiona que los esfuerzos mínimos no se encuentren en los puntos a 0 y a 180 grados del concentrador sino en el apoyo de la placa, como se aprecia en la Figura 27.

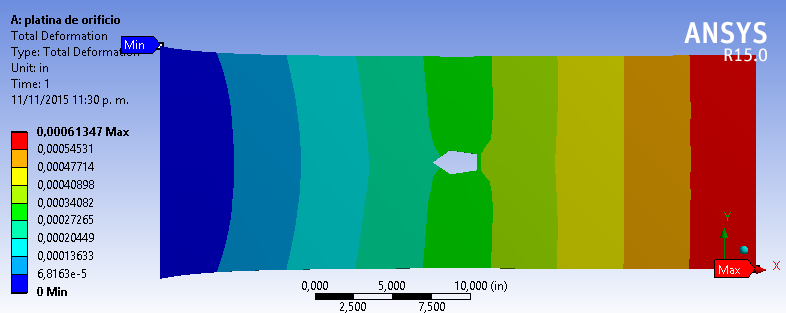


Figura 26. Deformaciones para malla gruesa.

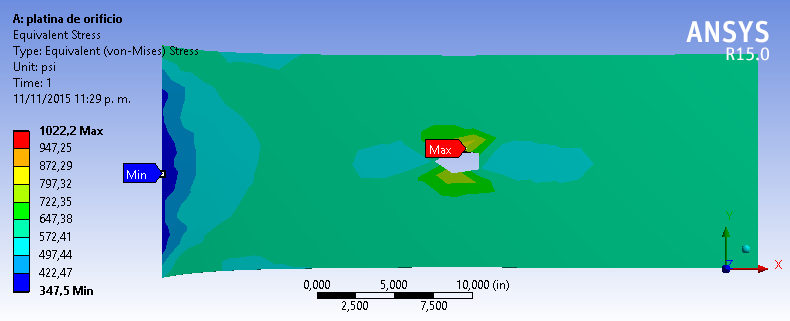


Figura 27. Esfuerzos para malla gruesa.

### Simulación realizada con malla fina:

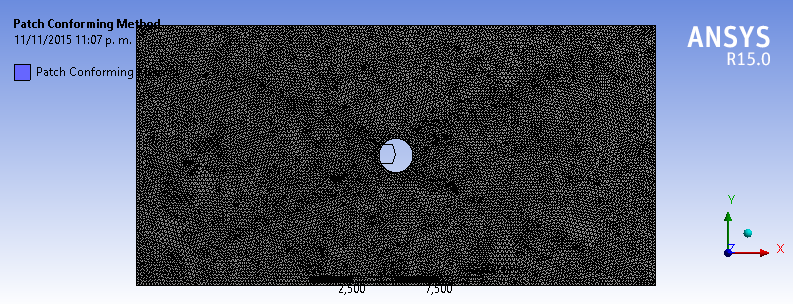
Simulación realizada elementos triangulares de tres nodos: 

Figura 28. Malla de 66670 Nodos y 305221 Elementos.

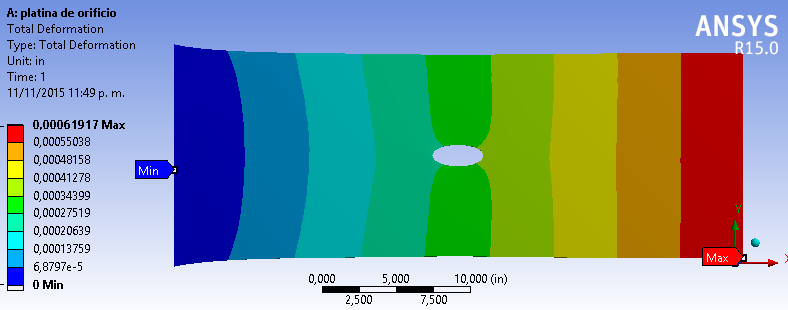
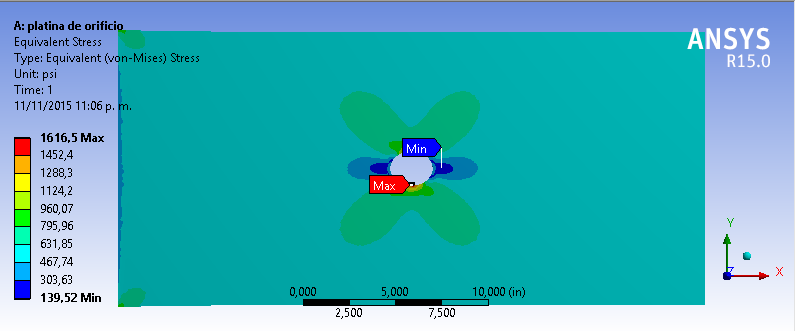


Figura 29. Deformaciones para malla fina.

Como puede observarse cuando la malla se hace más fina, se simula mediante los triángulos de forma más homogénea el concentrador de esfuerzos, permitiendo que los resultados de los esfuerzos máximos y mínimos se encuentren en los puntos correctos, 0° y 180° para los mínimos, y 90° y 270° para los máximos.

Figura 30 Esfuerzos para malla fina.

### Simulación realizada con malla intermedia:



Figura 31 Distribución de malla de 4627 Nodos y 17464 elementos triangulares de tres nodos.

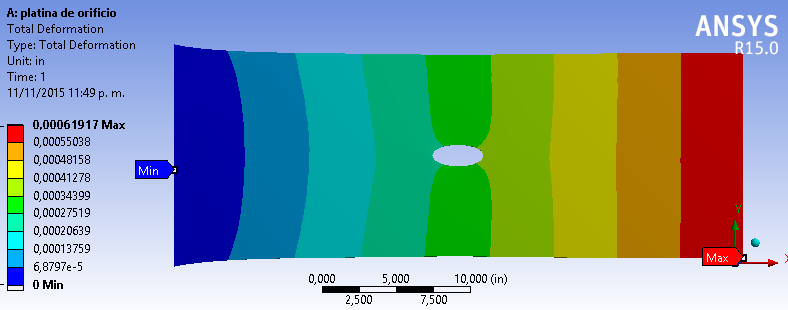


Figura 32 Deformaciones para malla intermedia.

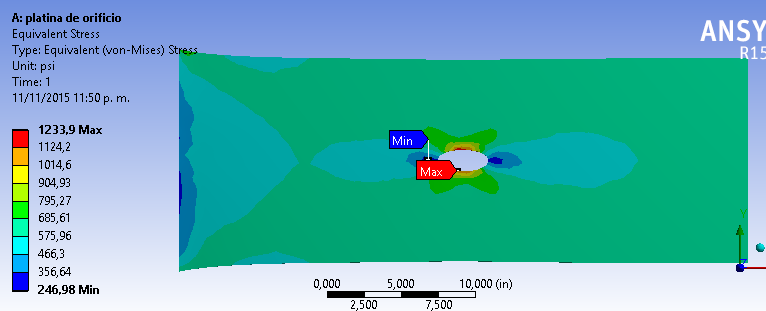


Figura 33 Esfuerzos para malla intermedia.

### Simulación realizada con malla intermedia con elementos triangulares de 6 nodos.

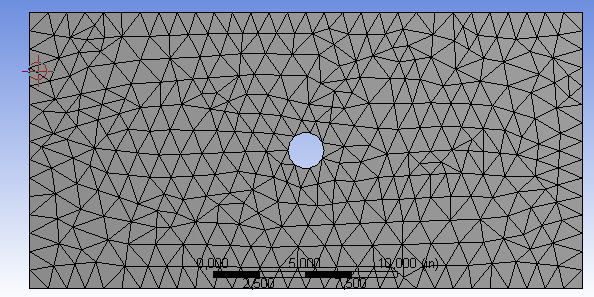


Figura 34 Distribución de malla de 4627 Nodos y 17464 elementos triangulares de seis nodos.

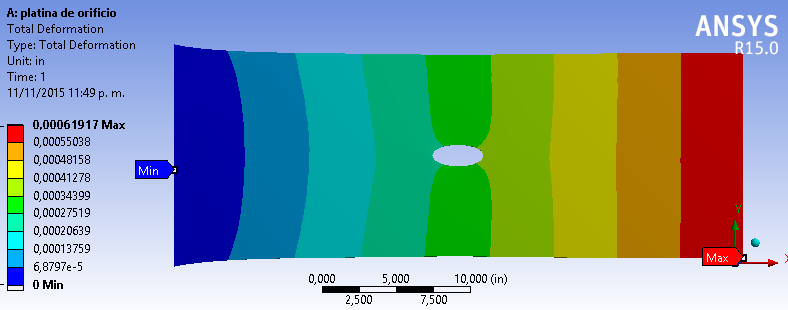


Figura 35 Deformaciones para malla intermedia elementos triangulares de seis nodos.

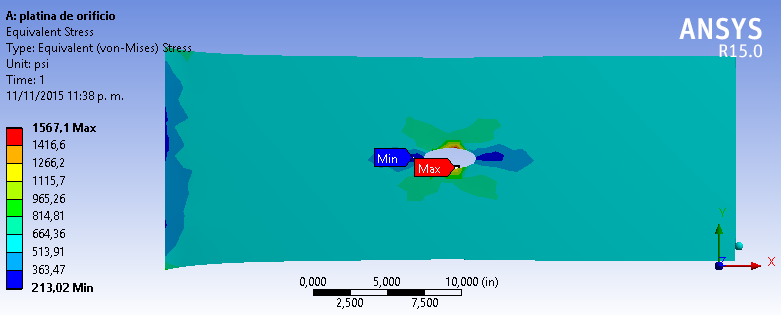


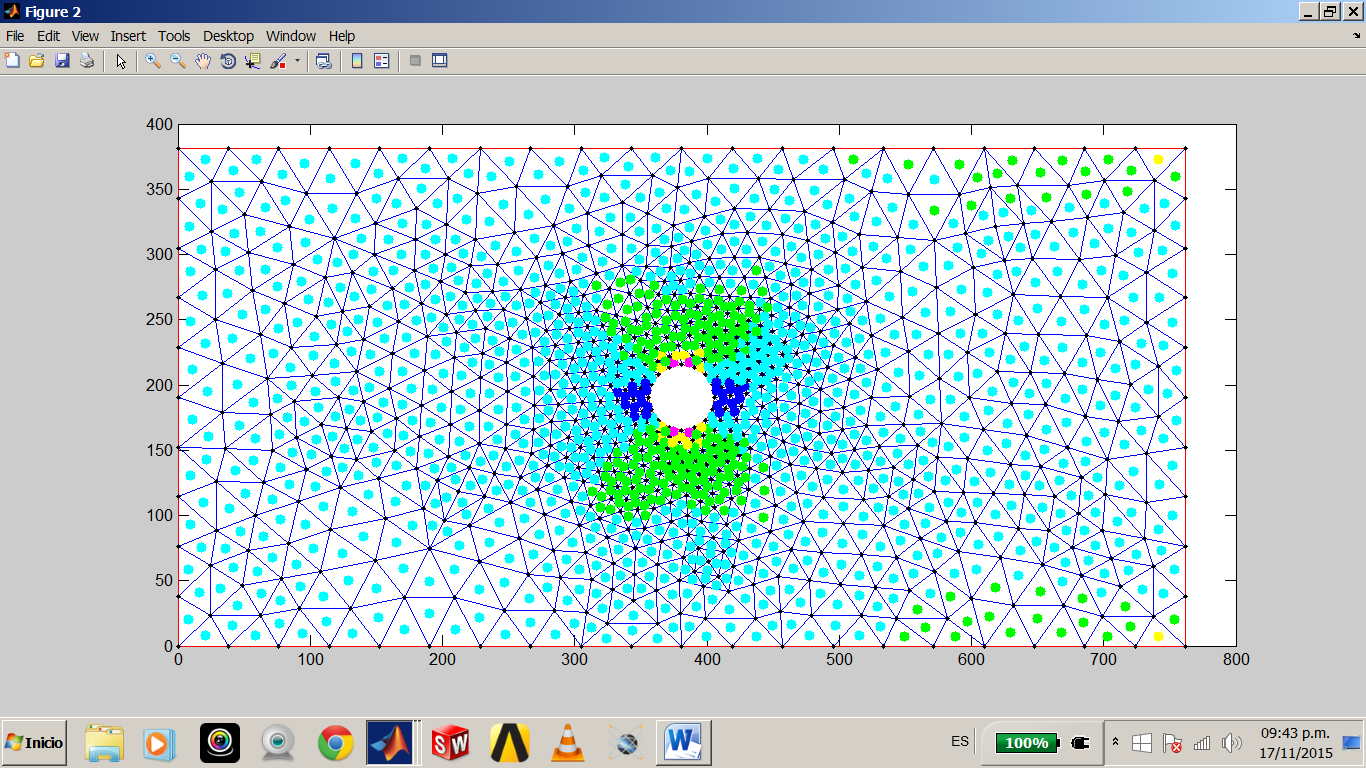
Figura 36 Esfuerzo para malla intermedia elementos triangulares de seis nodos.

El valor de referencia para comparación tomado del cálculo teórico para el esfuerzo máximo del diseño de la placa y correspondiente a la formulación de concentradores de esfuerzos mediante las curvas de Peterson es de 1799 psi.

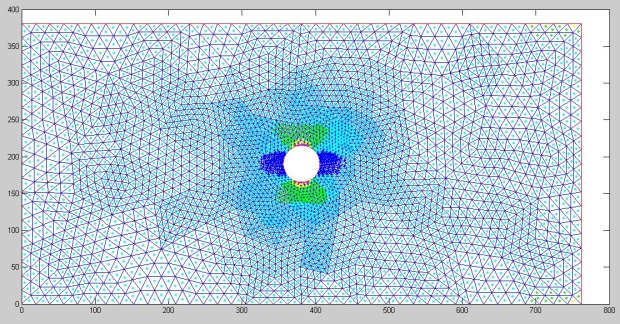
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **# de Nodos** | **# elementos** | **Valor Esfuerzo máximo(psi)** | **Valor esfuerzo mínimo (psi)** | **%Error para esfuerzo máximo** |
| **985** | 2674 | 1022.2 | 347.5 | 43% |
| **4627** | 17464 | 1233.9 | 247 | 31% |
| **66670** | 305221 | 1616.5 | 139.52 | 10% |
| **Análisis adicionando en cada triangulo 6 nodos** | | | | |
| **4621** | 2182 | 1567.1 | 213 | 13% |

Tabla 1. Tabla Comparativa de Deformaciones y Esfuerzos obtenidos.

## **RESULTADOS OBTENIDOS CON ALGORITMO IMPLEMENTADO.**



a)



b)

Figura 37 a) ESFUERZO EN X con maya de 1288elementos,

b) ESFUERZO EN X con maya de 5152 elementos.

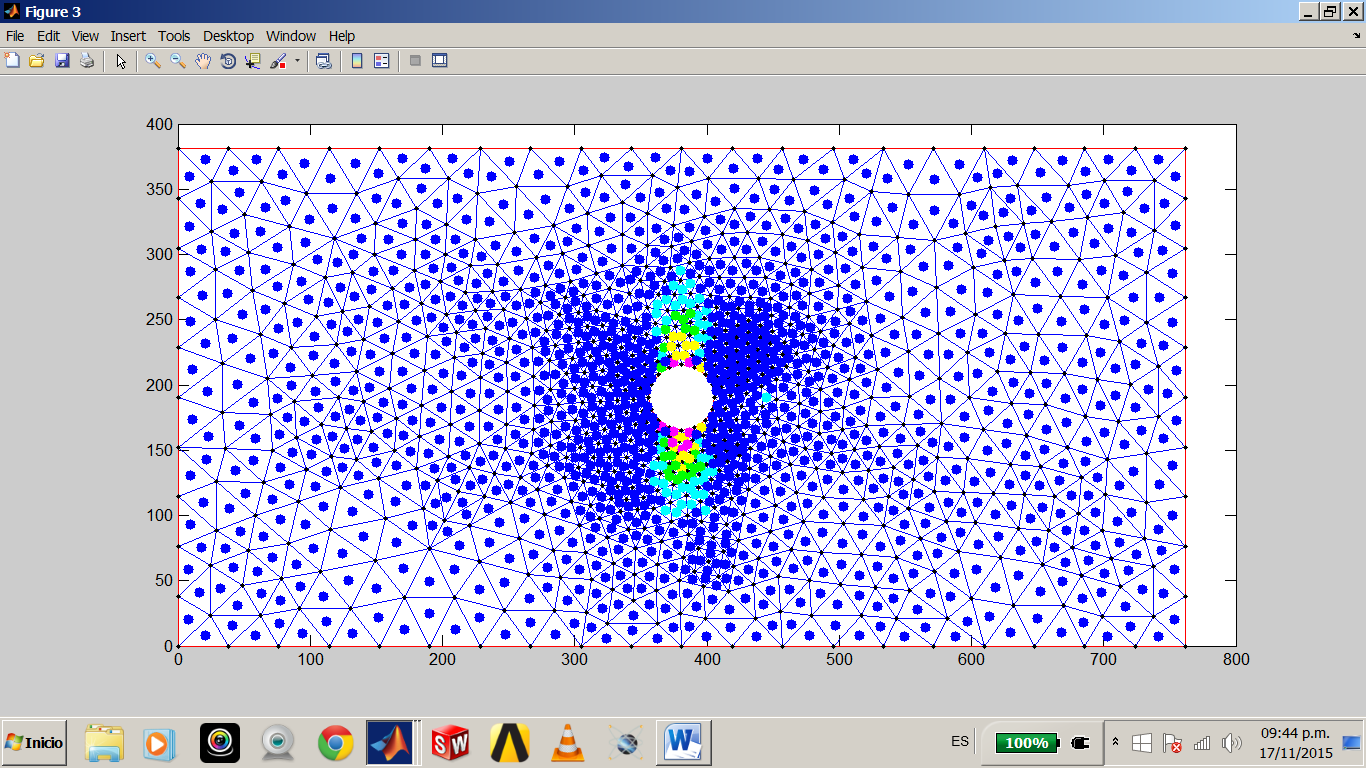


Figura 38 ESFUERZO EN Y

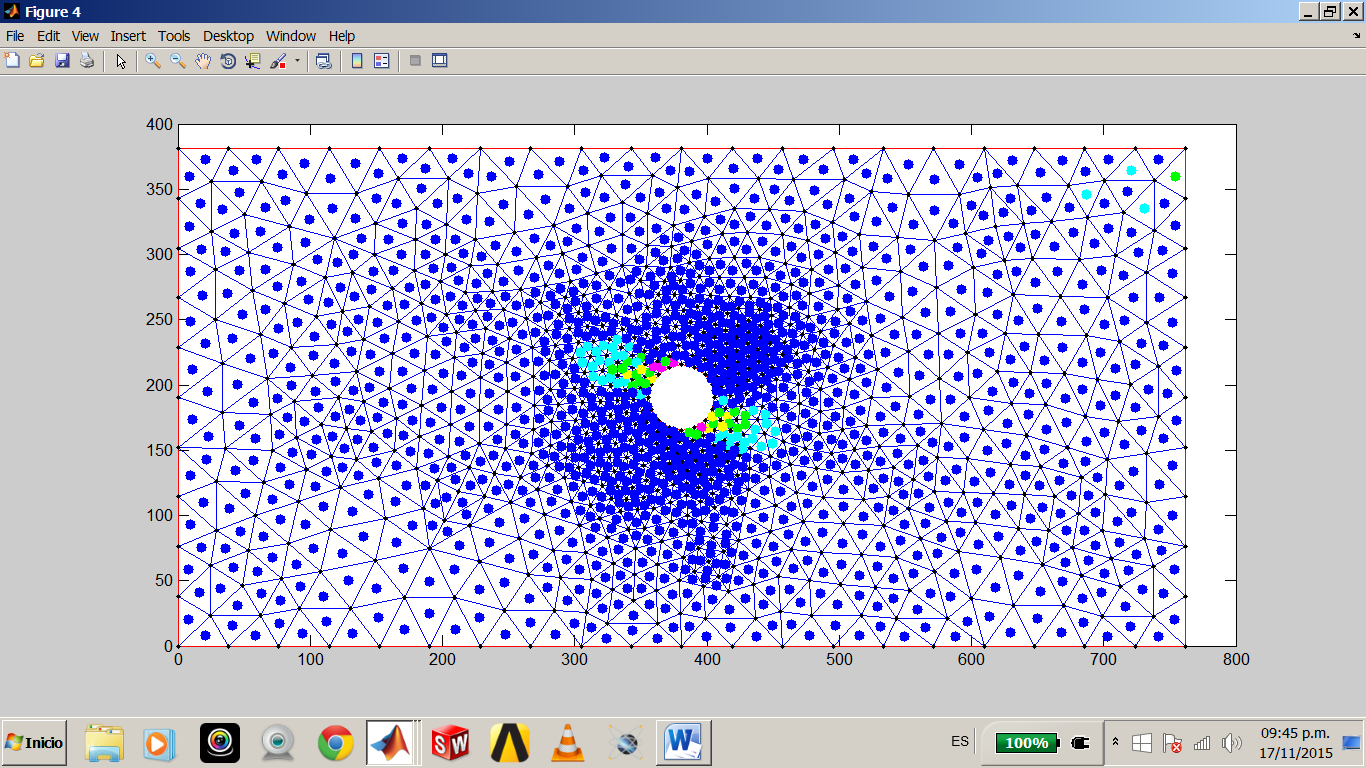


Figura 39 ESFUERZO EN XY

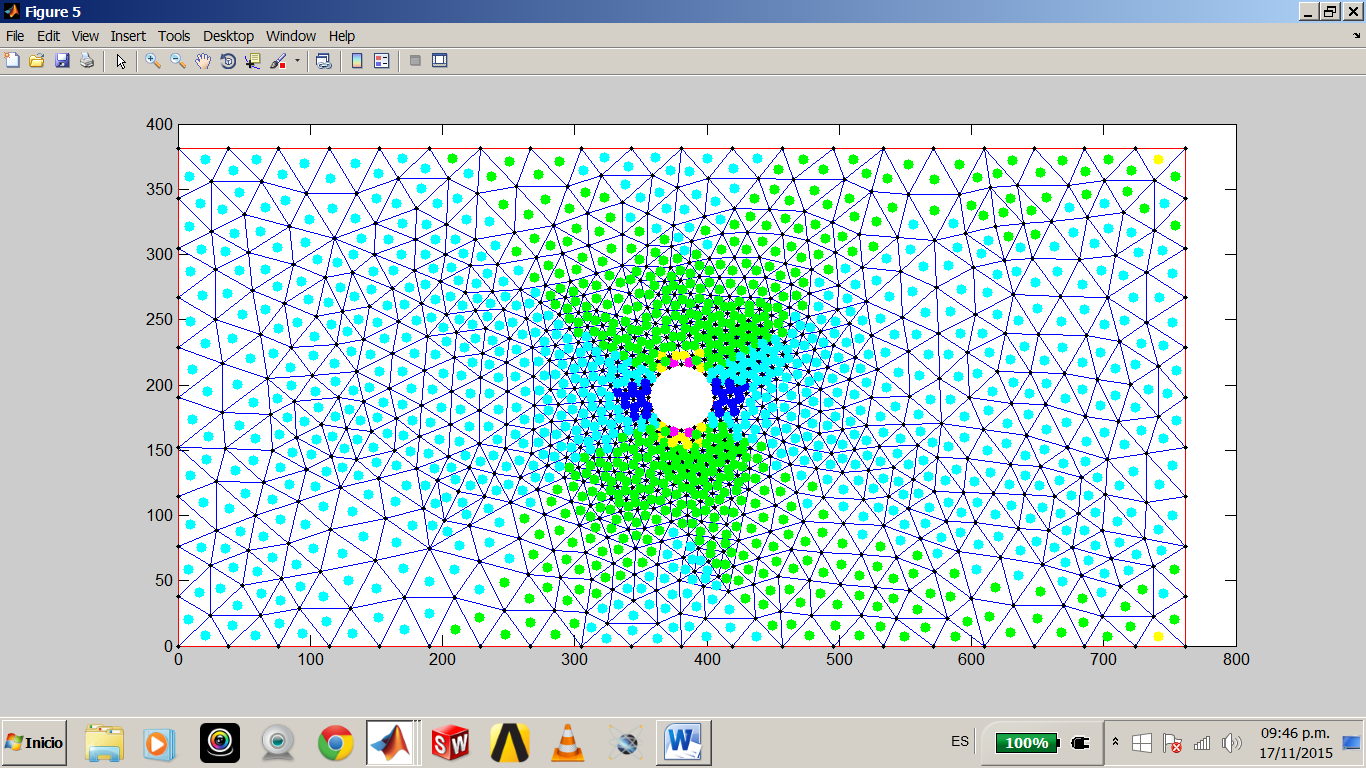


Figura 40 DEFORMACION EN X

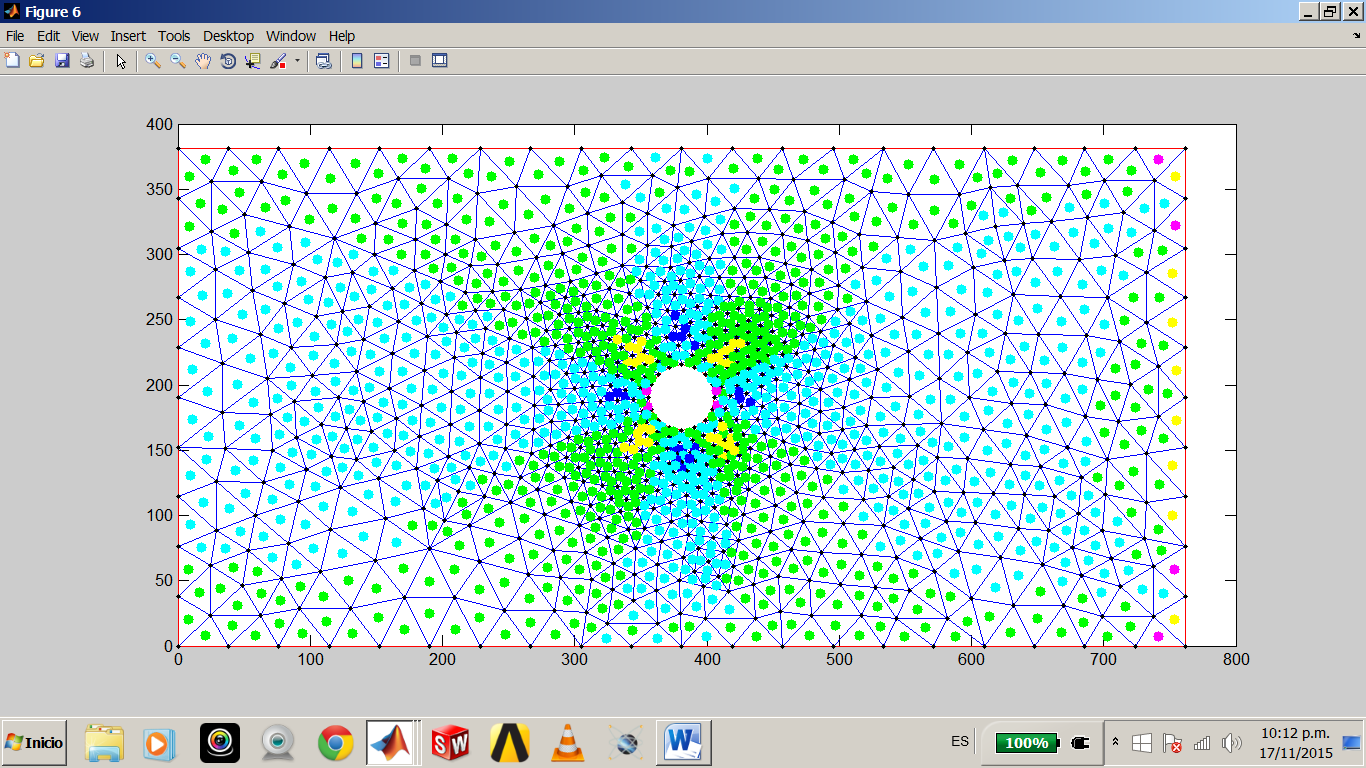


Figura 41 DEFORMACION EN Y

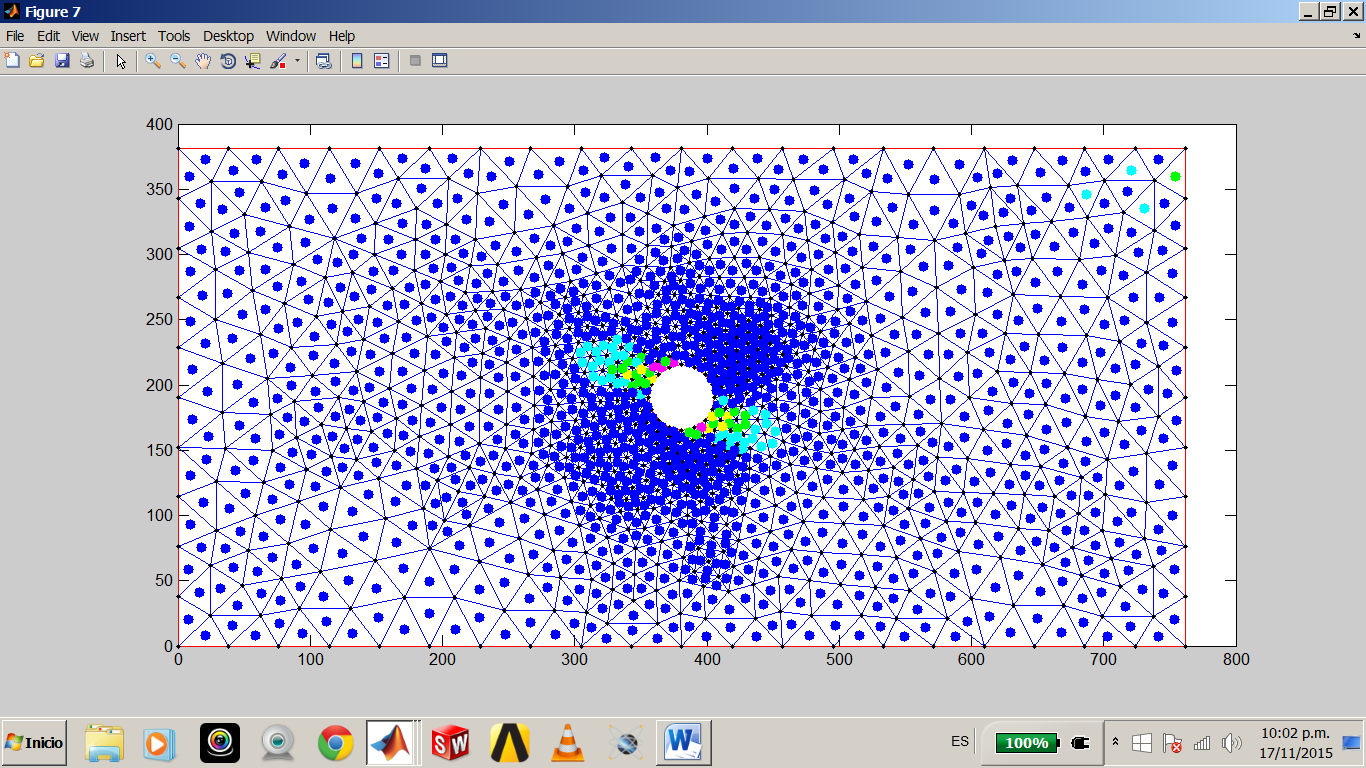


Figura 42 DEFORMACION EN XY.

ALGORITMOS

Función en MATLAB para la creación de la malla.

|  |
| --- |
| % This script is written and read by pdetool and should NOT be edited.  % There are two recommended alternatives:  % 1) Export the required variables from pdetool and create a MATLAB script  % to perform operations on these.  % 2) Define the problem completely using a MATLAB script. See  % http://www.mathworks.com/help/pde/examples/index.html for examples  % of this approach.    function cuadroef    [pde\_fig,ax]=pdeinit;  pdetool('appl\_cb',1);  set(ax,'DataAspectRatio',[220.60439560439556 142.72500000000002 1]);  set(ax,'PlotBoxAspectRatio',[1 1 1]);  set(ax,'XLim',[-10 772]);  set(ax,'YLim',[-10 391]);  set(ax,'XTickMode','auto');  set(ax,'YTickMode','auto');  pdetool('gridon','on');    % Geometry description:  pdeellip(381,190.5,25,25,...  0,'E1');  pdepoly([ 0,...  0,...  762,...  762,...  ],...  [ 0,...  381,...  381,...  0,...  ],...  'P1');  set(findobj(get(pde\_fig,'Children'),'Tag','PDEEval'),'String','P1-E1')    % Boundary conditions:  pdetool('changemode',0)  pdesetbd(5,...  'dir',...  1,...  '1',...  '0')  pdesetbd(4,...  'dir',...  1,...  '1',...  '0')  pdesetbd(3,...  'dir',...  1,...  '1',...  '0')  pdesetbd(2,...  'dir',...  1,...  '1',...  '0')  pdesetbd(1,...  'dir',...  1,...  '1',...  '0')    % Mesh generation:  setappdata(pde\_fig,'Hgrad',1.3);  setappdata(pde\_fig,'refinemethod','regular');  setappdata(pde\_fig,'jiggle',char('on','mean',''));  setappdata(pde\_fig,'MesherVersion','preR2013a');    pdetool('initmesh');  pdetool('refine')  % pdetool('refine')  % pdetool('refine')      % PDE coefficients:  pdeseteq(1,...  '1.0',...  '0.0',...  '10.0',...  '1.0',...  '0:10',...  '0.0',...  '0.0',...  '[0 100]')    setappdata(pde\_fig,'currparam',...  ['1.0 ';...  '0.0 ';...  '10.0';...  '1.0 '])    % Solve parameters:  setappdata(pde\_fig,'solveparam',...  char('0','4608','10','pdeadworst',...  '0.5','longest','0','1E-4','','fixed','Inf'))    % Plotflags and user data strings:  setappdata(pde\_fig,'plotflags',[1 1 1 1 1 1 1 1 0 0 0 1 1 0 0 0 0 1]);  setappdata(pde\_fig,'colstring','');  setappdata(pde\_fig,'arrowstring','');  setappdata(pde\_fig,'deformstring','');  setappdata(pde\_fig,'heightstring','');  end |

Algoritmo para el cálculo de matrices para el desarrollo del MEF.

|  |
| --- |
| close all    %FUNCION ENCARGADA DE CALCULAR LA MALLA CON ELEMENTOS TRIAGULARES Y NODOS  cuadroef3()    sp= size(p);  %NUMERO DE NODOS  nodos=sp(2);    %NUMERO DE DIMENSIONES  dimensiones = 2;    st= size(t);    % NUMERO DE ELEMENTOS  elementos = st(2);    % NUMERO DE GRADOS DE LIBERTAD  ngrad=dimensiones\*nodos;    %GRAFICACION DE MALLA  figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')    figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')    figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')      figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')      figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')    figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')      figure  pdemesh(p,e,t)  hold on  %MATRIZ DE COORDENADAS NODALES  Mcoorde= p';  %GRAFICACION DE NODOS  plot(Mcoorde(:,1),Mcoorde(:,2),'.k')      %MATRIZ DE CONECTIVIDAD  tt= t';  Mconectividad= tt(:,1:3);    %MATRIZ DE GRADOS DE LIBERTAD DOF  Mdof=zeros(elementos,6);  i=1;    while i<elementos+1    Mdof(i,1)=2\*Mconectividad(i,1)-1;  Mdof(i,2)=2\*Mconectividad(i,1);  Mdof(i,3)=2\*Mconectividad(i,2)-1;  Mdof(i,4)=2\*Mconectividad(i,2);  Mdof(i,5)=2\*Mconectividad(i,3)-1;  Mdof(i,6)=2\*Mconectividad(i,3);    i=i+1;  end    %MATRIZ DE PROPIEDADES  Eyoung = (2\*10^11)/(1000^2);  Vpoison = 0.3;  Tespesor = 12;    KK=zeros(ngrad);    D =(Eyoung/(1- (Vpoison^2))).\*[ 1 Vpoison 0; Vpoison 1 0; 0 0 (1-Vpoison)/2];    i = 1; %CONTADOR DE ELEMENTOS    %MATRIZ KK  while i <elementos+1    bi= Mcoorde(Mconectividad(i,2),2)- Mcoorde(Mconectividad(i,3),2); %yi-yk  bj= Mcoorde(Mconectividad(i,3),2)- Mcoorde(Mconectividad(i,1),2); %yk-yi  bk= Mcoorde(Mconectividad(i,1),2)- Mcoorde(Mconectividad(i,2),2); %yi-yj    ci = Mcoorde(Mconectividad(i,3),1)- Mcoorde(Mconectividad(i,2),1);%xk-xj  cj = Mcoorde(Mconectividad(i,1),1)- Mcoorde(Mconectividad(i,3),1);%xi-xk  ck = Mcoorde(Mconectividad(i,2),1)- Mcoorde(Mconectividad(i,1),1);%xj-xi    dosA = det([1 Mcoorde(Mconectividad(i,1),1) Mcoorde(Mconectividad(i,1),2);1 Mcoorde(Mconectividad(i,2),1) Mcoorde(Mconectividad(i,2),2);1 Mcoorde(Mconectividad(i,3),1) Mcoorde(Mconectividad(i,3),2)]);    Bb= [bi 0 bj 0 bk 0;0 ci 0 cj 0 ck; ci bi cj bj ck bk];    B=(1/dosA).\*Bb;    Btras= B';    Ke= Btras\*D\*B.\*Tespesor.\*(dosA/2);    a= Mdof(i,1);  b= Mdof(i,2);  c= Mdof(i,3);  d= Mdof(i,4);  e= Mdof(i,5);  f= Mdof(i,6);    KK(a,a)=KK(a,a)+Ke(1,1);  KK(a,b)=KK(a,b)+Ke(1,2);  KK(a,c)=KK(a,c)+Ke(1,3);  KK(a,d)=KK(a,d)+Ke(1,4);  KK(a,e)=KK(a,e)+Ke(1,5);  KK(a,f)=KK(a,f)+Ke(1,6);    KK(b,a)=KK(b,a)+Ke(2,1);  KK(b,b)=KK(b,b)+Ke(2,2);  KK(b,c)=KK(b,c)+Ke(2,3);  KK(b,d)=KK(b,d)+Ke(2,4);  KK(b,e)=KK(b,e)+Ke(2,5);  KK(b,f)=KK(b,f)+Ke(2,6);    KK(c,a)=KK(c,a)+Ke(3,1);  KK(c,b)=KK(c,b)+Ke(3,2);  KK(c,c)=KK(c,c)+Ke(3,3);  KK(c,d)=KK(c,d)+Ke(3,4);  KK(c,e)=KK(c,e)+Ke(3,5);  KK(c,f)=KK(c,f)+Ke(3,6);    KK(d,a)=KK(d,a)+Ke(4,1);  KK(d,b)=KK(d,b)+Ke(4,2);  KK(d,c)=KK(d,c)+Ke(4,3);  KK(d,d)=KK(d,d)+Ke(4,4);  KK(d,e)=KK(d,e)+Ke(4,5);  KK(d,f)=KK(d,f)+Ke(4,6);    KK(e,a)=KK(e,a)+Ke(5,1);  KK(e,b)=KK(e,b)+Ke(5,2);  KK(e,c)=KK(e,c)+Ke(5,3);  KK(e,d)=KK(e,d)+Ke(5,4);  KK(e,e)=KK(e,e)+Ke(5,5);  KK(e,f)=KK(e,f)+Ke(5,6);    KK(f,a)=KK(f,a)+Ke(6,1);  KK(f,b)=KK(f,b)+Ke(6,2);  KK(f,c)=KK(f,c)+Ke(6,3);  KK(f,d)=KK(f,d)+Ke(6,4);  KK(f,e)=KK(f,e)+Ke(6,5);  KK(f,f)=KK(f,f)+Ke(6,6);    i=i+1;  end    %BUSQUEDA DE NODOS DE BORDES PARA APLICACION DE FUERZAS EN TENSION  posicionborde= find( Mcoorde(:,1)== 762);    posicionbordeinicial= find( Mcoorde(:,1)== 0);  % posicionbase= find( Mcoorde(:,2)== 0);    numnodosfuerza=size(posicionborde);  numnodosfuerza=numnodosfuerza(1);    %CALCULO DE FUERZA NODAL POR APLICAR  Fnodal= (50\*381)/numnodosfuerza;    FF=zeros(ngrad,1);  gradosliberxf = (2.\*posicionborde)-1;  i=1;  while i< numnodosfuerza+1  FF(gradosliberxf(i))= FF(gradosliberxf(i))+Fnodal;  i=i+1;  end    gradosliberx0 = (2.\*posicionbordeinicial)-1;    %%% CORRESPONDE TODOS LOS U QUE TIENEN DESPLAZAMIENTO CERO%%%%  gradoslibereliminar= gradosliberx0;    uu= size(gradoslibereliminar);  uelimi= uu(1);    Ugradosliber = ones(ngrad,1);  i=1;    while i< uelimi+1  Ugradosliber(gradoslibereliminar(i))=0;  i=i+1;  end    tamanoredu = numel(Ugradosliber(Ugradosliber==1));  desco = find(Ugradosliber);  KKREDU = zeros(tamanoredu);    i=1;  while i<tamanoredu+1  j=1;  while j<tamanoredu+1  KKREDU(i,j)= KK(desco(i),desco(j));  j=j+1;  end  i=i+1;  end    FFREDU = zeros(tamanoredu,1);  i=1;  while i<tamanoredu+1  FFREDU(i,1)= FF(desco(i));  i=i+1;  end    Usol= KKREDU\FFREDU;    UTOTAL= zeros(ngrad,1);  i=1;  while i<tamanoredu+1  UTOTAL(desco(i),1)= Usol(i);  i=i+1;  end    DEFORMACIONTOTAL=zeros(elementos,3);  ESFUERZOTOTAL=zeros(elementos,3);    i=1;  while i <elementos+1    bi= Mcoorde(Mconectividad(i,2),2)- Mcoorde(Mconectividad(i,3),2); %yj-yk  bj= Mcoorde(Mconectividad(i,3),2)- Mcoorde(Mconectividad(i,1),2); %yk-yi  bk= Mcoorde(Mconectividad(i,1),2)- Mcoorde(Mconectividad(i,2),2); %yi-yj    ci = Mcoorde(Mconectividad(i,3),1)- Mcoorde(Mconectividad(i,2),1);%xk-xj  cj = Mcoorde(Mconectividad(i,1),1)- Mcoorde(Mconectividad(i,3),1);%xi-xk  ck = Mcoorde(Mconectividad(i,2),1)- Mcoorde(Mconectividad(i,1),1);%xj-xi    dosA =det([1 Mcoorde(Mconectividad(i,1),1) Mcoorde(Mconectividad(i,1),2);1 Mcoorde(Mconectividad(i,2),1) Mcoorde(Mconectividad(i,2),2);1 Mcoorde(Mconectividad(i,3),1) Mcoorde(Mconectividad(i,3),2)]);    Bb= [bi 0 bj 0 bk 0;0 ci 0 cj 0 ck; ci bi cj bj ck bk];    Belemental=(1/dosA).\*Bb;    j=1;  Uelemental=zeros(6,1);  while j<6+1  Uelemental(j,1)= UTOTAL(Mdof(i,j));  j=j+1;  end    deformacionelemental= Belemental\*Uelemental;    DEFORMACIONTOTAL(i,1)=deformacionelemental(1);  DEFORMACIONTOTAL(i,2)=deformacionelemental(2);  DEFORMACIONTOTAL(i,3)=deformacionelemental(3);    esfuerzoelemental= D\*deformacionelemental;    ESFUERZOTOTAL(i,1)=esfuerzoelemental(1);  ESFUERZOTOTAL(i,2)=esfuerzoelemental(2);  ESFUERZOTOTAL(i,3)=esfuerzoelemental(3);    i=i+1;    end |

Algoritmo para el ploteo de los gráficos de las deformaciones y esfuerzos obtenidos.

|  |
| --- |
| %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ESFUERZO EN X  c = 1;    esfuerzonormax = ESFUERZOTOTAL(:,1)/max(ESFUERZOTOTAL(:,1));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if esfuerzonormax(c,1)<= 0.2  figure(2)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if esfuerzonormax(c,1)>0.2 && esfuerzonormax(c,1)<= 0.4  figure(2)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if esfuerzonormax(c,1)>0.4 && esfuerzonormax(c,1)<= 0.6  figure(2)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if esfuerzonormax(c,1)>0.6 && esfuerzonormax(c,1)<= 0.8  figure(2)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if esfuerzonormax(c,1)>0.8 && esfuerzonormax(c,1)<= 1  figure(2)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ESFUERZO EN Y  c=1;  esfuerzonormay = ESFUERZOTOTAL(:,2)/max(ESFUERZOTOTAL(:,2));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if esfuerzonormay(c,1)<= 0.2  figure(3)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if esfuerzonormay(c,1)>0.2 && esfuerzonormay(c,1)<= 0.4  figure(3)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if esfuerzonormay(c,1)>0.4 && esfuerzonormay(c,1)<= 0.6  figure(3)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if esfuerzonormay(c,1)>0.6 && esfuerzonormay(c,1)<= 0.8  figure(3)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if esfuerzonormay(c,1)>0.8 && esfuerzonormay(c,1)<= 1  figure(3)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ESFUERZO EN XY  c=1;  esfuerzonormaxy = ESFUERZOTOTAL(:,3)/max(ESFUERZOTOTAL(:,3));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if esfuerzonormaxy(c,1)<= 0.2  figure(4)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if esfuerzonormaxy(c,1)>0.2 && esfuerzonormaxy(c,1)<= 0.4  figure(4)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if esfuerzonormaxy(c,1)>0.4 && esfuerzonormaxy(c,1)<= 0.6  figure(4)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if esfuerzonormaxy(c,1)>0.6 && esfuerzonormaxy(c,1)<= 0.8  figure(4)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if esfuerzonormaxy(c,1)>0.8 && esfuerzonormaxy(c,1)<= 1  figure(4)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DEFORMACION EN X  c=1;  defornormax = DEFORMACIONTOTAL(:,1)/max(DEFORMACIONTOTAL(:,1));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if defornormax(c,1)<= 0.2  figure(5)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if defornormax(c,1)>0.2 && defornormax(c,1)<= 0.4  figure(5)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if defornormax(c,1)>0.4 && defornormax(c,1)<= 0.6  figure(5)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if defornormax(c,1)>0.6 && defornormax(c,1)<= 0.8  figure(5)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if defornormax(c,1)>0.8 && defornormax(c,1)<= 1  figure(5)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DEFORMACION EN Y  c = 1;    defornormay = abs(DEFORMACIONTOTAL(:,2))/max(abs(DEFORMACIONTOTAL(:,2)));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if defornormay(c,1)<= 0.2  figure(6)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if defornormay(c,1)>0.2 && defornormay(c,1)<= 0.4  figure(6)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if defornormay(c,1)>0.4 && defornormay(c,1)<= 0.6  figure(6)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if defornormay(c,1)>0.6 && defornormay(c,1)<= 0.8  figure(6)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if defornormay(c,1)>0.8 && defornormay(c,1)<= 1  figure(6)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% DEFORMACION EN XY  c=1;  defornormaxy = DEFORMACIONTOTAL(:,3)/max(DEFORMACIONTOTAL(:,3));    while c < elementos+1    centro(c,:) = centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    if defornormaxy(c,1)<= 0.2  figure(7)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,0,1])  end      if defornormaxy(c,1)>0.2 && defornormaxy(c,1)<= 0.4  figure(7)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,1])  end      if defornormaxy(c,1)>0.4 && defornormaxy(c,1)<= 0.6  figure(7)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[0,1,0])  end    if defornormaxy(c,1)>0.6 && defornormaxy(c,1)<= 0.8  figure(7)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,1,0])  end    if defornormaxy(c,1)>0.8 && defornormaxy(c,1)<= 1  figure(7)  plot(centro(c,1),centro(c,2),'.','MarkerSize',20,'Color',[1,0,1])  end    c=c+1;    end    %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% PLOTEO DE NUMERO DE ELEMENTO  c = 1;    while c < elementos+1    centro(c,:) =centroid([Mcoorde(Mconectividad(c,1),1) Mcoorde(Mconectividad(c,1),2); Mcoorde(Mconectividad(c,2),1) Mcoorde(Mconectividad(c,2),2); Mcoorde(Mconectividad(c,3),1) Mcoorde(Mconectividad(c,3),2)]);    figure(1)    text(centro(c,1),centro(c,2),['',int2str(c),'']);    c=c+1;    end |

# CONCLUSIONES

Se pudo comprobar que los esfuerzos máximos ocurren en la parte central del concentrador de esfuerzo a 90° y 270° mediante el algoritmo realizado en MatLab lo que a su vez confirma la teoría de esfuerzos Saint Venant.

los resultados obtenidos para el cálculo de esfuerzos máximos mediante ANSYS con la malla fina, arrojo como resultado u valor de esfuerzo máximo de 11.1 N/mm2, mientras l valor obtenido por el algoritmo realizado con una malla fina es de 13.6329 N/mm2  tomando como referencia el valor dado por ANSYS el error es de un 18%.

Revisando los cálculos entregados por ANSYS con los diferentes tamaños de malla, y comparándolos con los resultados dados por el algoritmo, se evidencian errores entre 10 y 43% para el valor del esfuerzo máximo. Siendo el de menor porcentaje el obtenido con la malla más fina y el peor con la malla más gruesa, no obstante para comprobar la eficiencia de los elementos triangulares, se simulo un escenario de prueba con elementos triangulares de seis nodos arrojando un porcentaje de error entre 13%, lo que nos permite confirmar que elementos de mayor complejidad matemática en la funciones de forma arrojan resultados más precisos.

El enmallado permite crear el análisis por el método de elementos finitos, siempre y cuando se realice en orden y en forma anti horaria, además se pudo apreciar que los elementos que comparten nodos también comparten sus grados i, j y k- esimos de libertad.

Los máximos esfuerzos se encuentran ubicados en el concentrador de esfuerzos a 90° y 270°, y los esfuerzos mínimos en 0° y 180°.

# REFERENCIAS

1. INTRODUCCION AL ESTUDIO DEL ELEMENTO FINITO EN INGENIERIA.

Tirupathi R. Chandrupatla/ Ashok D.Belegundu Editorial PEARSON Prentice Hall.

1. SOFTWARE PARA CIENCIA E INGENIERIA MATLAB

Linder Amancio Rodríguez /Cristian Muñoz Villalobos Empresa Editora MACRO