



Titulación:	Grado en Tecnologías Industriales	Curso:	4º
Asignatura:	Métodos Matemáticos para Análisis de Modelos		
Actividad:	Resolución de un problema de elasticidad 3D utilizando FreeFem++		

1. Objetivo

Esta actividad se centra en la aplicación de los conceptos relacionados con la resolución mediante el método de elementos finitos del problema de elasticidad lineal tridimensional. Se utilizará el programa de código abierto GMSH para la creación de la geometría y de la malla del modelo. Para la resolución de la formulación variacional del problema se utilizará el programa FreeFem++. El objetivo de la práctica es un mejor entendimiento acerca de las herramientas de código abierto para la resolución de problemas de elasticidad lineal tridimensional.

La actividad será desarrollada de forma individual. A modo orientativo el alumno deberá completar las siguientes etapas:

1. Generar la geometría 3D con el programa GMSH.
2. Mallar dicha geometría con elementos tridimensionales y realizar la exportación de la malla en formato reconocible por FreeFem++ (*.msh).
3. Resolver del problema de elasticidad lineal:
 - Importar la malla obtenida con GMSH.
 - Definir las constantes reales y el espacio de elementos finitos.
 - Definir la formulación variacional del problema incluyendo las condiciones de contorno impuestas.
 - Resolución del sistema de ecuaciones resultante y representación del campo de desplazamientos.
 - Obtener y representar la tensión equivalente de von Mises en el dominio elástico.

2. Plazo de entrega

La entrega consistirá en un fichero comprimido con el siguiente contenido:

- Fichero *.geo con la geometría del modelo obtenida con GMSH.
- Fichero *.msh con la malla del modelo obtenida con GMSH.
- Código de FreeFem++.



2.1. Ejercicio

En este ejercicio se obtiene el campo de desplazamientos y tensiones de una biela. El dominio elástico está definido en la Figura 1. El material utilizado es acero con un módulo de elasticidad $E = 205,8e3$ MPa y un coeficiente de Poisson $\nu = 0,3$. La geometría se muestra en la Figura 1. La pieza está sometida a una carga total de 6400 N en dirección x.

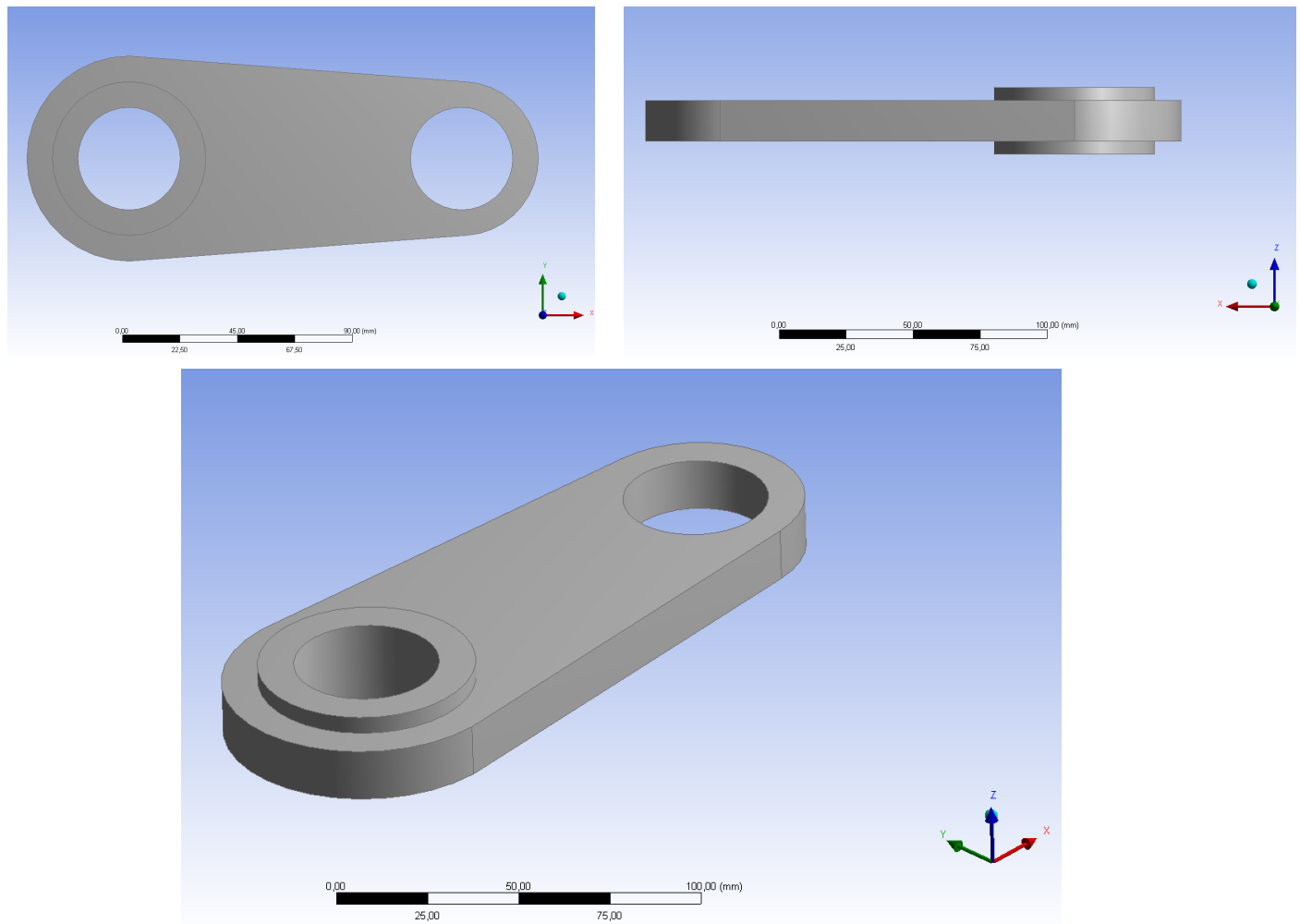


Figura 1. Geometría y condiciones de contorno.



2.1.1. Código GMSH

```
Point(1) = {0, 0, 0, 10};
Point(2) = {200, 0, 0, 10};
Point(3) = {10, 0, 0, 10};
Point(4) = {20, 0, 0, 10};
Point(5) = {40, 0, 0, 10};
Point(6) = {60, 0, 0, 10};
Point(7) = {70, 0, 0, 10};
Point(8) = {190, 0, 0, 10};
Point(9) = {170, 0, 0, 10};
Point(10) = {150, 0, 0, 10};
Point(11) = {170, 20, 0, 10};
Point(12) = {170, -20, 0, 10};
Point(13) = {170, 30, 0, 10};
Point(14) = {170, -30, 0, 10};
Point(15) = {40, 20, 0, 10};
Point(16) = {40, 30, 0, 10};
Point(17) = {40, 40, 0, 10};
Point(18) = {40, -20, 0, 10};
Point(19) = {40, -30, 0, 10};
Point(20) = {40, -40, 0, 10};
Line(1) = {13, 17};
Line(2) = {20, 14};
Circle(3) = {13, 9, 2};
Circle(4) = {2, 9, 14};
Circle(5) = {11, 9, 8};
Circle(6) = {8, 9, 12};
Circle(7) = {12, 9, 10};
Circle(8) = {10, 9, 11};
Circle(9) = {20, 5, 1};
Circle(10) = {1, 5, 17};
Circle(11) = {16, 5, 7};
Circle(12) = {7, 5, 19};
Circle(13) = {19, 5, 3};
Circle(14) = {3, 5, 16};
Circle(15) = {15, 5, 6};
Circle(16) = {6, 5, 18};
Circle(17) = {18, 5, 4};
Circle(18) = {4, 5, 15};
Line Loop(23) = {1, -10, -9, 2, -4, -3};
Line Loop(24) = {11, 12, 13, 14};
Line Loop(25) = {5, 6, 7, 8};
Plane Surface(26) = {23, 24, 25};
Line Loop(27) = {18, 15, 16, 17};
Plane Surface(28) = {24, 27};
Extrude {0, 0, 15} {
    Surface{26, 28};
}
Extrude {0, 0, 5} {
    Surface{142};
}
Extrude {0, 0, -5} {
    Surface{28};
}
Physical Volume(227) = {4, 2, 3, 1};
Physical Surface(229) = {141, 137, 133, 129};
Physical Surface(230) = {99, 95};
```



2.1.2. Código FreeFem

```
load "gmsh";

//***** PREPROCESO*****
// Cargamos la malla generada por gmsh
mesh3 Th = gmshload3("bielaSC.msh");
plot(Th,wait=true);

// Definir parámetros del problema unidades (N,mm)

real E=205.8e3; // N/mm2
real nu=0.3;
real Ftot=6400; // N
real Pres=Ftot/(3.1416*15*20);
real mu= E/(2*(1+nu)); // N/mm2
real lambda = E*nu/((1+nu)*(1-2*nu)); // N/mm2
real sqrt2=sqrt(2.);
real sigmay=250; // N/mm2

// Valor de parámetros por pantalla
cout << ""<<endl;
cout << "Coeficientes:"<<endl;
cout << "-----"<<endl;
cout << "E="<<E<<endl;
cout << "nu="<<nu<<endl;
cout << "Ftot="<<Ftot<<endl;
cout << "Pres="<<Pres<<endl;
cout << "mu="<<mu<<endl;
cout << "lambda="<<lambda<<endl;
cout << ""<<endl;

// Definir espacio de elementos finitos 3d
fespace Vh(Th,P2); // Lagrange P2
Vh u1,u2,u3,v1,v2,v3;
```

Figura 2. Código FreeFem++ (I): Etapa de Preproceso.

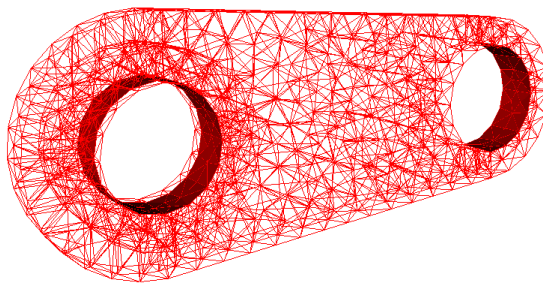


Figura 3. Malla de elementos finitos.



```
***** SOLUCION*****  
// macros para la formulacion variacional  
macro epsilon(u1,u2,u3) [dx(u1),dy(u2),dz(u3), (dz(u2)+dy(u3))/sqrt2,  
[dz(u1)+dx(u3))/sqrt2, (dy(u1)+dx(u2))/sqrt2] // EOM  
macro div(u1,u2,u3) ( dx(u1)+dy(u2)+dz(u3) ) // EOM  
  
// Formulación variacional y solución  
solve Lame([u1,u2,u3],[v1,v2,v3])=  
int3d(Th) {  
  lambda*div(u1,u2,u3)*div(v1,v2,v3)  
  +2.*mu*( epsilon(u1,u2,u3)'*epsilon(v1,v2,v3) ) //'  
}  
- int2d(Th,230) (Pres*v1) // Integrar presión en superficie 230  
+ on(229,u1=0,u2=0,u3=0) // gdl restringidos en superficie 229 (numerada en gmsh)  
;  
  
***** POSTPROCESO*****  
real dxmax = u1[].max; // calculamos el maximo desplazamiento en x  
real dymax = u2[].max; // calculamos el maximo desplazamiento en y  
real dzmax = u3[].max; // calculamos el maximo desplazamiento en z  
  
cout << "<<endl;  
cout << "Max. desplazamientos:"<<endl;  
cout << "-----"<<endl;  
cout << "dxmax="<<dxmax<<endl;  
cout << "dymax="<<dymax<<endl;  
cout << "dzmax="<<dzmax<<endl;  
  
plot(u1);
```

Figura 4. Código FreeFem++ (II): Formulación Variacional.

```
fespace Wh(Th,P1); // elementos P1  
Wh sigmavm;  
Wh FS;  
  
// un macro para las componentes del tensor de esfuerzos  
  
macro sigma(u1,u2,u3) [lambda*(dx(u1)+dy(u2)+dz(u3))+2*mu*dx(u1),  
lambda*(dx(u1)+dy(u2)+dz(u3))+2*mu*dy(u2),  
lambda*(dx(u1)+dy(u2)+dz(u3))+2*mu*dz(u3),  
mu*(dx(u1)+dy(u2)+dz(u3))] // EOM  
  
// definimos la tension de von Misses  
sigmavm = sqrt(sigma(u1,u2,u3)[0]*sigma(u1,u2,u3)[0]  
+sigma(u1,u2,u3)[1]*sigma(u1,u2,u3)[1]  
+sigma(u1,u2,u3)[2]*sigma(u1,u2,u3)[2]  
-sigma(u1,u2,u3)[0]*sigma(u1,u2,u3)[1]  
-sigma(u1,u2,u3)[1]*sigma(u1,u2,u3)[2]  
-sigma(u1,u2,u3)[2]*sigma(u1,u2,u3)[0]  
);  
  
plot(sigmavm);  
  
// definimos el factor de seguridad respecto al límite elastico del material  
FS=sigmavm/sigmay;
```

Figura 5. Código FreeFem++ (III): Cálculo de tensiones.



2.1.3. Resultados

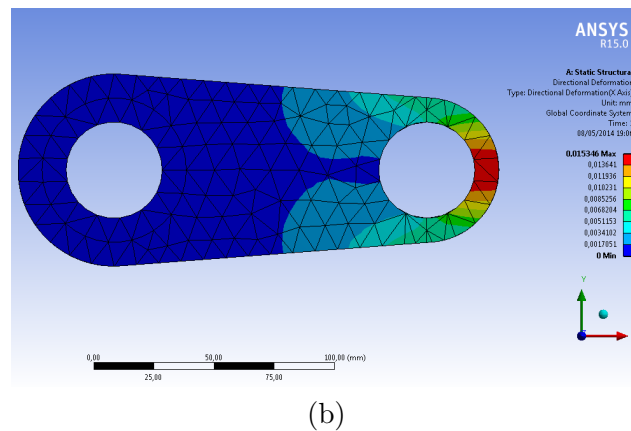
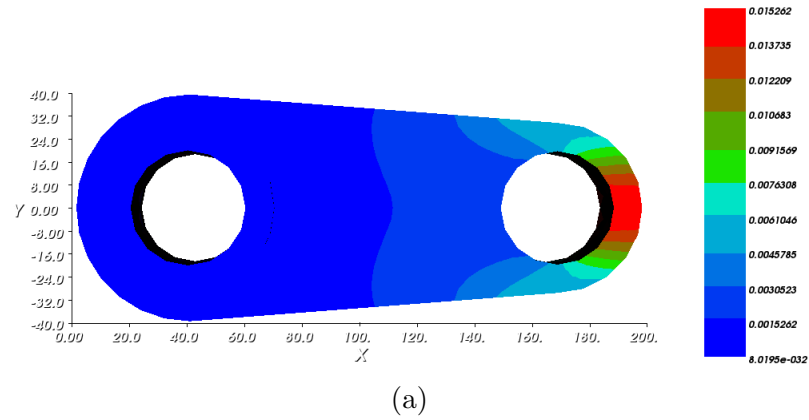
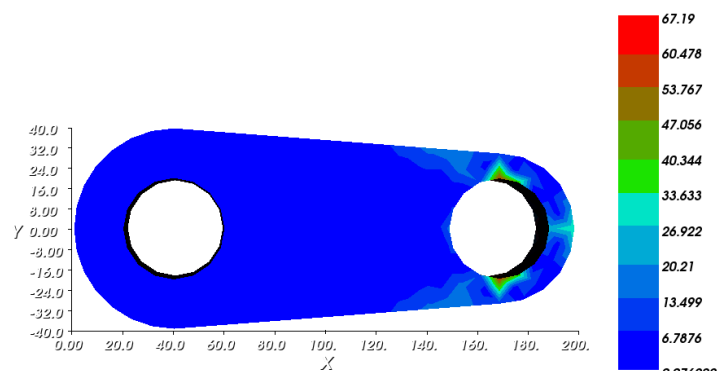
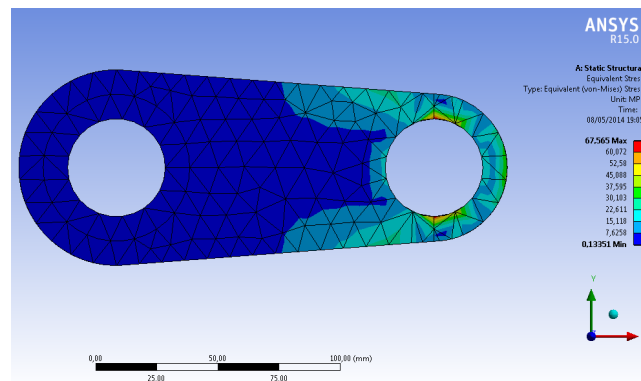


Figura 6. Desplazamiento en dirección x. (a) Solución con FreeFem++, (b) solución con ANSYS



(a)



(b)

Figura 7. Tensión de von Mises. (a) Solución con FreeFem++, (b) solución con ANSYS