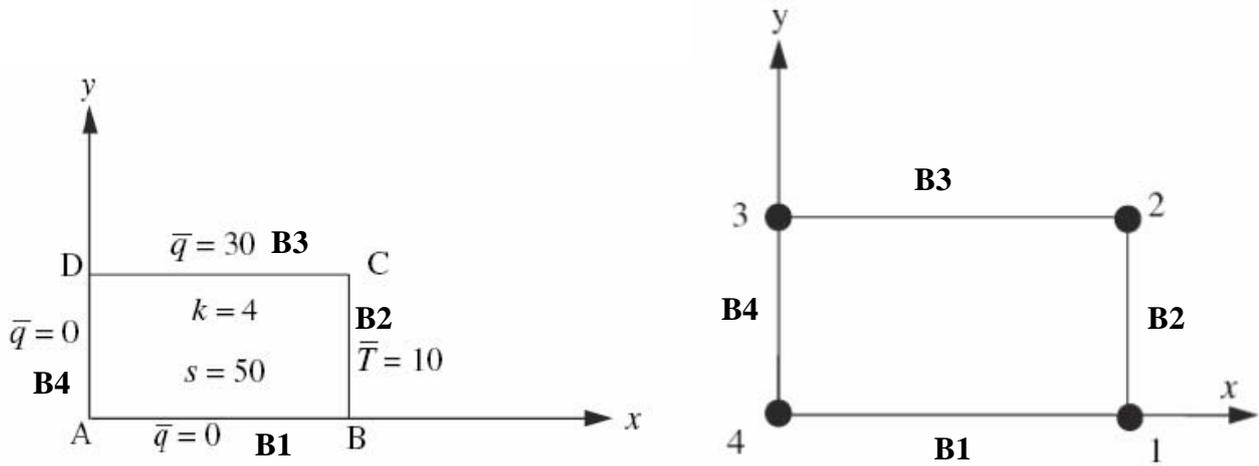


Prob PII-1. Conducción del calor en un dominio 2D rectangular (Cálculos a mano) (Fish&Belytschko, p210)



Se considera un problema de distribución de calor en un dominio rectangular (2x1 m) como se ve en la figura de la izquierda. La conductividad es $k=4 \text{ W}\cdot\text{°C}^{-1}$. La temperatura está forzada a $\bar{T} = 10^{\circ}\text{C}$ en el borde CB. Los lados AB y CD están aislados, es decir, en ellos $\bar{q} = 0 \text{ W}\cdot\text{m}^{-1}$. A lo largo del lado DC, el flujo de borde es $\bar{q} = 30 \text{ W}\cdot\text{m}^{-1}$. Sobre el rectángulo actúa una fuente distribuida de calor constante, $s=50 \text{ W m}^{-2}$

Obtener la temperatura en los nodos y los flujos en los nodos. Evaluar las matrices de los elementos mediante integración de Gauss. Utilizar un solo elemento finito rectangular, con la numeración de nodos indicada en la figura derecha, de modo que la numeración local y global coincida.

PII-1) Solución

La formulación o forma débil del problema se expresa así:

Encontrar una función $T(x,y) \in H^1(\Omega)$, con $T(s) = \bar{T}(s)$, $s \in \Gamma_T$ (Γ_T : zona del borde con condiciones esenciales, valores de T dados, Γ_q : zona del borde con condiciones esenciales, flujo q dado), tal que para cualquier función $w \in H^1(\Omega)$, con $w=0$ sobre Γ_T se verifica la siguiente ecuación (L10 p36):

$$\iint_{\Omega} k \cdot \nabla T \cdot \nabla w \cdot d\Omega = \iint_{\Omega} f \cdot w \cdot d\Omega - \int_{\Gamma_q} \bar{q} \cdot w \cdot d\Gamma$$

De la forma débil, se puede identificar fácilmente (L12 p32):

$$K^e = \iint_{\Omega^e} k \cdot B^T \cdot B \cdot d\Omega \quad , \quad f^e = \iint_{\Omega^e} N^T \cdot f \cdot d\Omega - \int_{\Gamma_q^e} N^T \cdot \bar{q} \cdot d\Gamma$$

Las funciones básicas en coordenadas naturales son (L13 p12):

$$N_1(x, y) = \hat{N}_1(\xi, \eta) = \frac{1}{4}(1 - \xi) \cdot (1 - \eta) \quad , \quad N_2(x, y) = \hat{N}_2(\xi, \eta) = \frac{1}{4}(1 + \xi) \cdot (1 - \eta)$$

$$N_3(x, y) = \hat{N}_3(\xi, \eta) = \frac{1}{4}(1 + \xi) \cdot (1 + \eta) \quad , \quad N_4(x, y) = \hat{N}_4(\xi, \eta) = \frac{1}{4}(1 - \xi) \cdot (1 + \eta)$$

De modo que la transformación es:

$$x = 2 \cdot N_1(x,y) + 2 \cdot N_2(x,y) = 1 - \eta$$

$$y = N_1(x,y) + N_3(x,y) = (1 + \xi) / 2$$

La matriz jacobiana resulta ser:

$$J^T = \begin{bmatrix} \frac{\partial x}{\partial \xi} & \frac{\partial y}{\partial \xi} \\ \frac{\partial x}{\partial \eta} & \frac{\partial y}{\partial \eta} \end{bmatrix} = \begin{bmatrix} 0 & 1/2 \\ -1 & 0 \end{bmatrix}$$

El determinante de la matriz jacobiana es: $|J|=1/2$

La inversa de la matriz jacobiana es: $J^{-1} = \begin{bmatrix} 0 & 2 \\ -1 & 0 \end{bmatrix}$

$$\begin{bmatrix} \frac{\partial N_1}{\partial \xi} & \frac{\partial N_2}{\partial \xi} & \frac{\partial N_3}{\partial \xi} & \frac{\partial N_4}{\partial \xi} \\ \frac{\partial N_1}{\partial \eta} & \frac{\partial N_2}{\partial \eta} & \frac{\partial N_3}{\partial \eta} & \frac{\partial N_4}{\partial \eta} \end{bmatrix} = \frac{1}{4} \begin{bmatrix} \eta-1 & 1-\eta & 1+\eta & -\eta-1 \\ \xi-1 & -1-\xi & 1+\xi & 1-\xi \end{bmatrix}$$

$$[B^e] = \begin{bmatrix} \frac{\partial \xi}{\partial x} & \frac{\partial \eta}{\partial x} \\ \frac{\partial \xi}{\partial y} & \frac{\partial \eta}{\partial y} \end{bmatrix} \begin{bmatrix} \frac{\partial \hat{N}_1^e}{\partial \xi} & \frac{\partial \hat{N}_2^e}{\partial \xi} & \frac{\partial \hat{N}_3^e}{\partial \xi} & \frac{\partial \hat{N}_4^e}{\partial \xi} \\ \frac{\partial \hat{N}_1^e}{\partial \eta} & \frac{\partial \hat{N}_2^e}{\partial \eta} & \frac{\partial \hat{N}_3^e}{\partial \eta} & \frac{\partial \hat{N}_4^e}{\partial \eta} \end{bmatrix} = \frac{1}{|J^e|} \begin{bmatrix} \frac{\partial y^e}{\partial \eta} & -\frac{\partial y^e}{\partial \xi} \\ -\frac{\partial x^e}{\partial \eta} & \frac{\partial x^e}{\partial \xi} \end{bmatrix} \begin{bmatrix} \frac{\partial \hat{N}_1^e}{\partial \xi} & \frac{\partial \hat{N}_2^e}{\partial \xi} & \frac{\partial \hat{N}_3^e}{\partial \xi} & \frac{\partial \hat{N}_4^e}{\partial \xi} \\ \frac{\partial \hat{N}_1^e}{\partial \eta} & \frac{\partial \hat{N}_2^e}{\partial \eta} & \frac{\partial \hat{N}_3^e}{\partial \eta} & \frac{\partial \hat{N}_4^e}{\partial \eta} \end{bmatrix}$$

Esto resulta de una deducción anterior

$$\begin{Bmatrix} d\xi \\ d\eta \end{Bmatrix} = \frac{1}{|J^e|} \begin{bmatrix} \frac{\partial y^e}{\partial \eta} & -\frac{\partial x^e}{\partial \eta} \\ -\frac{\partial y^e}{\partial \xi} & \frac{\partial x^e}{\partial \xi} \end{bmatrix} \begin{Bmatrix} dx \\ dy \end{Bmatrix}$$

$$B = J^{T^{-1}} \begin{bmatrix} \frac{\partial \hat{N}_1}{\partial \xi} & \frac{\partial \hat{N}_2}{\partial \xi} & \frac{\partial \hat{N}_3}{\partial \xi} & \frac{\partial \hat{N}_4}{\partial \xi} \\ \frac{\partial \hat{N}_1}{\partial \eta} & \frac{\partial \hat{N}_2}{\partial \eta} & \frac{\partial \hat{N}_3}{\partial \eta} & \frac{\partial \hat{N}_4}{\partial \eta} \end{bmatrix} = \frac{1}{4} \cdot \begin{bmatrix} 1-\xi & 1+\xi & -1-\xi & \xi-1 \\ 2(\eta-1) & 2(1-\eta) & 2(1+\eta) & -2(\eta+1) \end{bmatrix}$$

La matriz de rigidez es:

$$K^e = \iint_{\Omega^e} k \cdot B^T \cdot B \cdot d\Omega = \int_{-1}^1 \int_{-1}^1 k \cdot B^T \cdot B \cdot |J| \cdot d\xi \cdot d\eta =$$

$$\{\text{Gauss-Leg}\} = \sum_{i=1}^2 \sum_{j=1}^2 k \cdot (B^T \cdot B \cdot |J|)_{\xi=\xi_i, \eta=\eta_j} \cdot W_i \cdot W_j =$$

$$= \begin{bmatrix} 3.3333 & -2.3333 & -1.6667 & 0.6667 \\ -2.3333 & 3.3333 & 0.6667 & -1.6667 \\ -1.6667 & 0.6667 & 3.3333 & -2.3333 \\ 0.6667 & -1.6667 & -2.3333 & 3.3333 \end{bmatrix}$$

La contribución de la 'fuente' es:

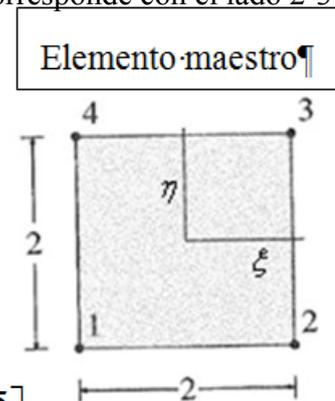
$$f_{\Omega}^e = \int_{-1}^1 \int_{-1}^1 f \cdot N^T \cdot |J| \cdot d\xi \cdot d\eta =$$

$$= (50/2) \int_{-1}^1 \int_{-1}^1 \begin{Bmatrix} \frac{1}{4} \cdot (1 - \xi) \cdot (1 - \eta) \\ \frac{1}{4} \cdot (1 + \xi) \cdot (1 - \eta) \\ \frac{1}{4} \cdot (1 + \xi) \cdot (1 + \eta) \\ \frac{1}{4} \cdot (1 - \xi) \cdot (1 + \eta) \end{Bmatrix} \cdot d\xi \cdot d\eta = 25 \cdot \begin{Bmatrix} \left(\frac{1}{4}\right) \cdot 2 \cdot 2 \\ \left(\frac{1}{4}\right) \cdot 2 \cdot 2 \\ 1 \\ 1 \end{Bmatrix} = \begin{Bmatrix} 25 \\ 25 \\ 25 \\ 25 \end{Bmatrix}$$

Ahora hay que calcular la contribución a la 'carga' del elemento de las condiciones de borde naturales a lo largo del lado CD. Considerando coordenadas 'naturales' ξ - η , ese lado se corresponde con el lado 2-3 del elemento 'maestro'. Para él se tiene: $\xi=1$, η varía de -1 a 1

$$f_{\Omega}^e = - \int_{-1}^1 \bar{q} \cdot (N^T)_{\xi=1} \cdot \underbrace{|j|}_{\text{factor cambio var 1D}} \cdot d\eta =$$

$$= -30 \int_{-1}^1 \begin{Bmatrix} 0 \\ \frac{1}{2} \cdot (1 - \eta) \\ \frac{1}{2} \cdot (1 + \eta) \\ 0 \end{Bmatrix} \frac{2-0}{2} \cdot d\eta = \begin{Bmatrix} 0 \\ -30 \\ -30 \\ 0 \end{Bmatrix}$$



El ensamblado de todas las contribuciones de los nodos da:

$$\begin{bmatrix} 3.3333 & -2.3333 & -1.6667 & 0.6667 \\ -2.3333 & 3.3333 & 0.6667 & -1.6667 \\ -1.6667 & 0.6667 & 3.3333 & -2.3333 \\ 0.6667 & -1.6667 & -2.3333 & 3.3333 \end{bmatrix} \begin{bmatrix} T_1 = 10 \\ T_2 = 10 \\ T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} r_1 + 25 \\ r_2 - 5 \\ -5 \\ 25 \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} 3.3333 & -2.3333 \\ -2.3333 & 3.3333 \end{bmatrix} \begin{bmatrix} T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} -5 \\ 25 \end{bmatrix} - \begin{bmatrix} -1.6667 & 0.6667 \\ 0.6667 & -1.6667 \end{bmatrix} \begin{bmatrix} 10 \\ 10 \end{bmatrix} = \begin{bmatrix} 5 \\ 35 \end{bmatrix} \Rightarrow$$

$$\begin{bmatrix} T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} 17.3529 \\ 22.6471 \end{bmatrix} \text{ } ^\circ\text{C}$$

Los flujos de calor en los nodos son:

$$\{q^1(\xi, \eta)\} = \begin{bmatrix} -k \frac{\partial T}{\partial x} \\ -k \frac{\partial T}{\partial y} \end{bmatrix} = -k [B^1] \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix}$$

$$\{q^1(\xi, \eta)\} = -4 \begin{bmatrix} \frac{1 - \xi}{4}, & \frac{1 + \xi}{4}, & -\frac{(1 + \xi)}{4}, & -\frac{(1 - \xi)}{4} \\ -\frac{(1 - \eta)}{2}, & \frac{(1 - \eta)}{2}, & \frac{1 + \eta}{2}, & -\frac{(1 + \eta)}{2} \end{bmatrix} \begin{bmatrix} 10 \\ 10 \\ 17 \\ 385 \end{bmatrix}$$

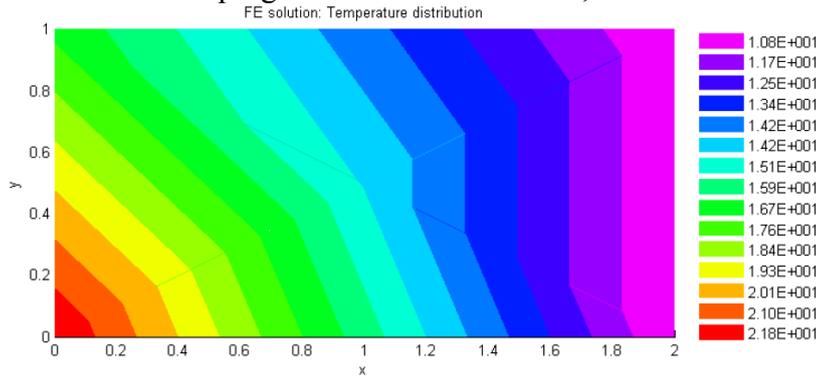
Nodo 1: $(\xi, \eta) = (-1, -1)$, Nodo 2: $(\xi, \eta) = (1, -1)$, Nodo 3: $(\xi, \eta) = (1, 1)$, Nodo 4: $(\xi, \eta) = (-1, 1)$

Nodo 1: $q^1(-1, -1) = [430/17, 0]$, Nodo 2: $q^1(1, -1) = [250/17, 0]$

Nodo 3: $q^1(1, 1) = [250/17, 360/17]$, Nodo 4: $q^1(-1, 1) = [430/17, 360/17]$

Las temperaturas en los nodos son
$$\begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{pmatrix} = \begin{pmatrix} 10 \\ 10 \\ 295/17 \\ 385/17 \end{pmatrix}$$

Comparando con la solución del programa 2dBVP en Matlab, se obtienen los mismos resultados:

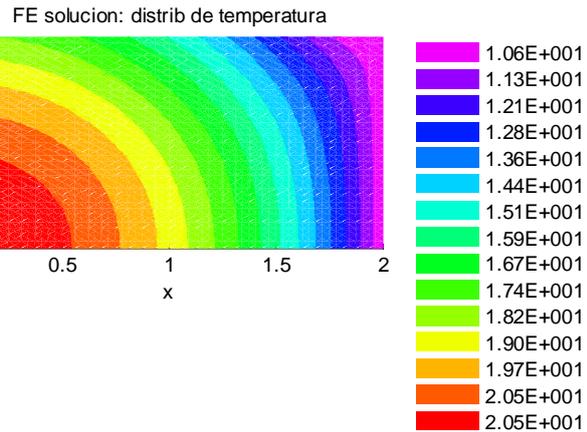
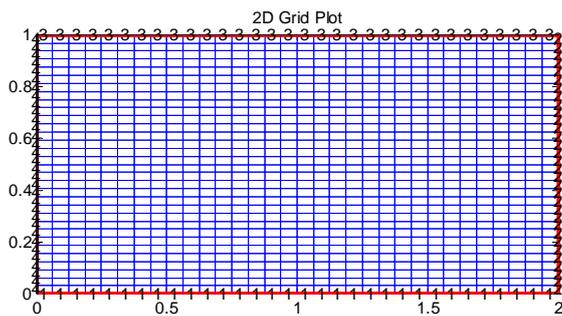


$d = [10 \quad 10 \quad 17.353 \quad 22.647]'$

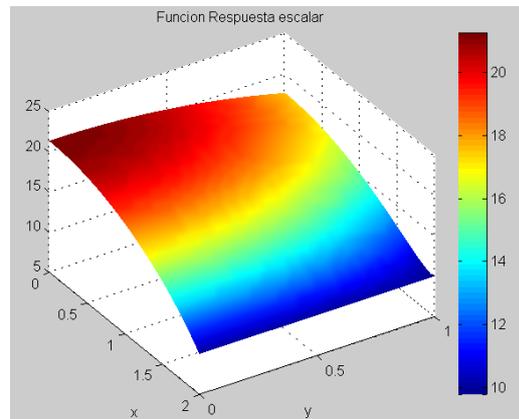
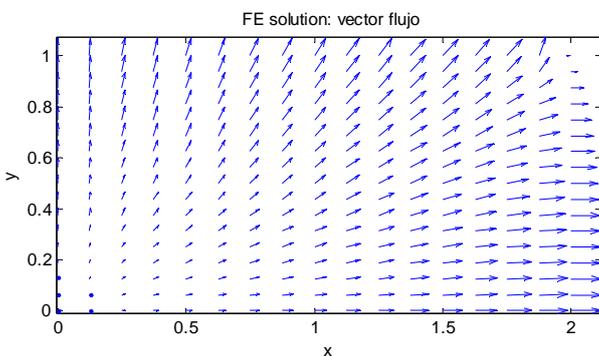
Tras 5 refines de la malla, l tabla de evolución de temperaturas en los nodos iniciales resulta:

Evol Temperaturas en nodos 1, 2, 3, 4 tras 5 refines (1024 elem, 1089 nodos):

1	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000
2	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000
3	17.35294	17.73793	17.54229	17.51439	17.50785	17.50622
4	22.64706	21.41140	21.28704	21.25514	21.24703	21.24500



Flujo tras 4 refines

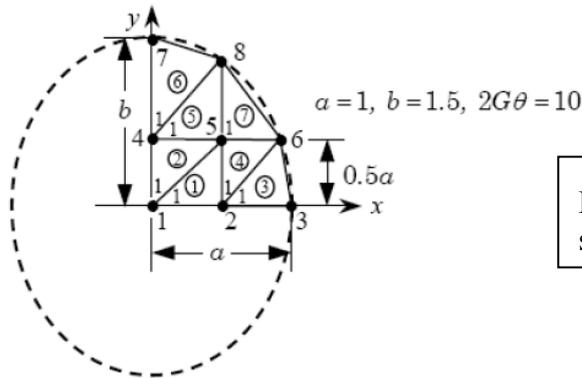


Con malla triangular inicial de 2 elementos

Evol de Temperaturas en nodos 1, 2, 3, 4 tras 6 refines (8192 elem, 4225 nodos):

1	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000
2	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000	10.00000
3	19.25926	18.12260	17.71914	17.57419	17.52644	17.51177	17.50743
4	20.74074	20.94573	21.10918	21.19536	21.22845	21.23946	21.24288

Problema PII-2. Torsión de una pieza cilíndrica de sección constante



Por simetría, se puede utilizar un cuadrante de la sección como dominio computacional

La teoría de Prandtl de la torsión de una pieza cilíndrica lleva a la EDP:

$$-\nabla^2 u = 2 \cdot G \cdot \theta \quad \text{en } \Omega, \quad u=0 \text{ sobre } \Gamma \text{ (el borde de la sección } \Omega)$$

donde Ω es la sección transversal de la pieza cilíndrica sometida a torsión, Γ es la curva contorno de Ω , G es el módulo de corte del material e la pieza, θ es el ángulo de giro en la torsión, y u es la función de tensión.

SE PIDE:

Resolver la ecuación para el caso en que Ω una sección elíptica de la pieza, utilizando la malla de elementos triangulares lineales o de elementos cuadriláteros implementada en el programa 2dBVP en Matlab.

Comparar la solución por elementos finitos con la solución exacta (válida para secciones elípticas con ejes a y b)

$$u = \frac{G \cdot \theta \cdot a^2 \cdot b^2}{a^2 + b^2} \left(1 - \frac{x^2}{a^2} - \frac{y^2}{b^2} \right)$$

Utilizar $a=1$, $b=1.5$, y $f_0=2 \cdot G \cdot \theta=10$

Obsérvese que la malla mostrada en la figura es sólo para ilustrar. Por simetría, se utilizará un cuarto del dominio.

Se podría crear una malla con algún software. Si se utiliza ANSYS se puede utilizar la función loadFromGridFile.m del programa 2dBVP para leer la malla. Es interesante comprobar la convergencia de la malla.

Se utilizarán funciones de mallado en formulación de coordenadas polares (función FunGeomPolar.m en carpeta 2dBVP_EF_MallasPolares) así como las modificaciones introducidas al respecto en los diversos módulos de 2dBVP (Main2D_refine, InputIniGeometria, InputData, include_variables,...). Observar la evolución de la convergencia a lo largo de sucesivos refines de la malla d elementos finitos.

Una vez que la función de tensión u se ha obtenido, las tensiones se puede calcular de las expresiones:

$$\sigma_{xz} = G \cdot \theta \cdot \frac{\partial u}{\partial y} \quad , \quad \sigma_{yz} = - G \cdot \theta \cdot \frac{\partial u}{\partial x}$$

Dibujar el vector tensión: $\mathbf{v} = \bar{\sigma}_{xz} \mathbf{i} + \bar{\sigma}_{yz} \mathbf{j}$, donde $\bar{\sigma}_{xz} = \sigma_{xz} / (G \cdot \theta)$, $\bar{\sigma}_{yz} = \sigma_{yz} / (G \cdot \theta)$

PII-2. Solución

Se señalan modificaciones en el programa 2dBVP en Matlab. Algunas se refieren al programa básico y otras a módulos modificados para considerar mallas de elementos cuadriláteros utilizando coordenadas polares.

Para este problema hay que modificar los archivos generales exact.m y ff.m para preparar el tratamiento de partida del problema.

Se han de modificar InputData.m para asignar condiciones esenciales y naturales de los contornos, que se pueden considerar de diversos modos. (curvo). Las funciones specifyEssBCValue.m y specifyNaturalBValue.m se debe modificar de acuerdo con lo anterior.

Para utilizar el mallado generado por el programa ANSYS, se debe modificarla función Preprocessor.m se que el programa importe la malla de los correspondientes archivos ANSYS, que deben colocarse en el mismo directorio del disco que los otros archivos Matlab del programa.

Para mallado con polares se ha de modificar la función InputIni_Geometria.m , llamada por InputGrid_refine.m, y utilizar FunGeomPolar.m. Se han de utilizar versiones adaptadas para mallado con polares de otras funciones (Main2D_refine, InputIniGeometria, InputData, include_variables,...)

La función postprocessor.m (postprocessor_refine.m) se ha de modificar para que se dibujen las variables deseadas.

exact.m

Introducir en esta función ,la solución analítica exacta y sus derivadas parciales:

```
a=1;
b=1.5;
GTheta=5;
u=((5*a^2*b^2)/(a^2+b^2))*(1-(x(1)^2/a^2)-(x(2)^2/b^2));
dudx=(-2*x(1)/a^2)*((5*a^2*b^2)/(a^2+b^2));
dudy=(-2*x(2)/b^2)*((5*a^2*b^2)/(a^2+b^2));
```

ff.m

Aplicar una 'fuerza' distribuida en el dominio de $f_0=2G\theta=10$

```
value = 10;
```

InputData.m

Asignar condiciones de contorno o borde

. En la malla ANSYS se toma como borde B1 el curvo, en el que se aplica condición esencial, y en los bordes B2 y B3, sobre los ejes de simetría de la sección, se toman condiciones naturales nulas. Por ello:

```
IsBoundaryCondition = [1 0 0];
```

. En la solución con mallado polar, el borde curvo se ha llamado B3, sin otra razón que cambiar la nomenclatura. En los B1 y B2 sobre los ejes de simetría, hay condiciones naturales nulas, la proyección del gradiente sobre la normal vale cero.

```
IsBoundaryCondition = [0,0,1];
```

specifyEssBCValue.m

. Si se usa ANSYS: se toma como borde B1 el curvo elíptico y para indicar que se aplica en él una condición de contorno esencial de 0:

```
if ( sideInd == 1)
    dof = ndof*(nodeID - 1) + 1;
    vals = 0;
end
```

. Si se usa mallado polar (función FunGeompolar)

```
% if ( sideInd == 3) % Caso torsion polares, u es cero en contorno
%     dof = ndof*(nodeID - 1) + 1; % Here, we only have one dof equal to
%     vals = 0; % vals are scalars for this program!
% end
```

specifyNaturalBCValue.m

No hay condiciones de contorno naturales no nulas en este problema, por tanto:

```
alpha = 0; % initialize the two output variables
beta = 0;
```

Conductivity.m

Hay que especificar valores correctos en la matriz de 'conductividad':

```
D=[1,0;0,1];
```

preprocessor.m , preprocessor_refine.m

. Si se usa malla ANSYS, hay que pedir al programa que cargue la malla de ANSYS

```
load_grid = 'yes'; % en función preprocessor_refine;
```

. Si se usa proceso de refino progresivo de malla inicial, se debe indicar en preprocessor_refine.m

```
load_grid = 'refine';
```

postprocessor.m, postprocessor_refine.m

Dibujar el vector de tensión especificado:

```
plotvector(+gradient(:,2),-gradient(:,1),'FE solution: Stress Vector',0.9);
plotvector(grad(:,2),-grad(:,1),'Exact solution: Stress Vector',0.9);
```

InputIniGeometria.m

Esta función para definir la geometría y su evolución, es llamada por InputGrid_refine.m, que es llamada por preprocessor_refine.m, y ésta por el programa principal Main2D_refine.m .

En la gestión de mallado con polares, hay algunas peculiaridades.

```
% Datos geom secc elipt torsion
nir=4;exm=0;exM=1;eym=0;eyM=1.5;% en el refines solo cambia nir, se duplica
na=5;am=0;aM=pi/2; % en el refine solo cambia (na-1)num subinterv ang, se duplica
FunGeomPolar % funcion genera malla 'polar'
nno=size(Nodes,2);% n° nodos
nno_ini=nno; % num nodos primera iteracion
nel=size(Elms,2); % n° elem
```

Explicaciones:

La llamada a FunGeomPolar.m no lleva argumentos ni variables 'respuesta', pues todas ellas son de tipo global, así introducidas en la función include_variables.m, algo modificada para el mallado con polares.

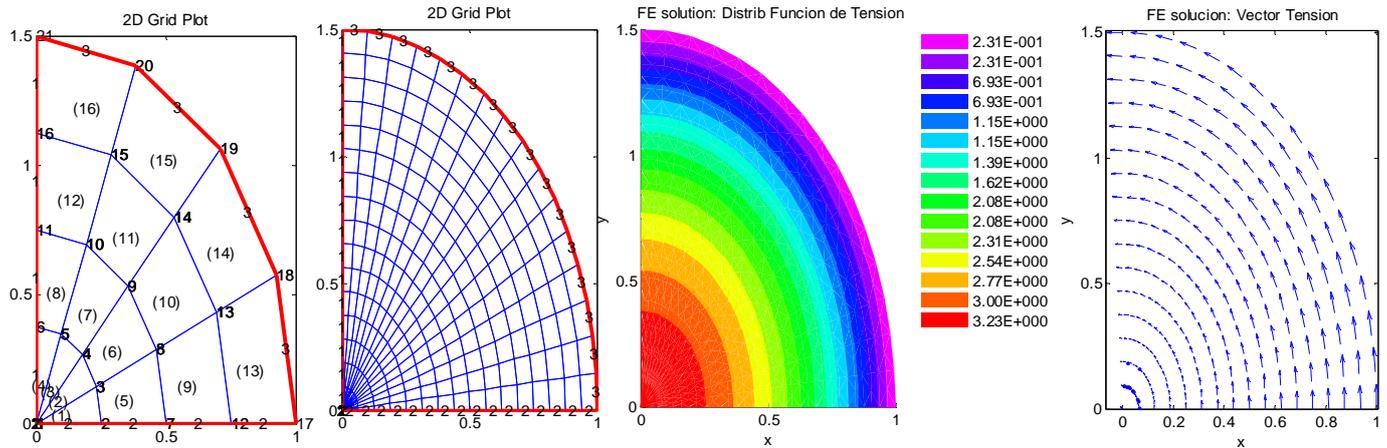
Para el mallado inicial, sobre el dominio que es un cuarto de interior de elipse, se toma:

```
nir=4;exm=0;exM=1;eym=0;eyM=1.5;
```

. 4 intervalos entre elipses, es decir, el punto origen (0,0) y 4 grupos de puntos sobre 4 elipses en el mallado inicial. Los ejes mínimo y máximo de las elipses varía, desde el origen (exm=0,eym=0) hasta la más exterior (exm=1, eym=1.5)

. se consideran na=5 'radiovectores' para ubicar nodos, desde el origen, en ángulos recorridos de derecha a izquierda, desde el mínimo am=0, al máximo aM=pi/2.

El refine del mallado tras la ejecución del mallado inicial, se realiza por bipartición del numero de intervalos entre elipses y del número de ángulos recorridos. En cada refine se cuadruplica (2x2) el número de elementos cuadriláteros. Los elementos en torno al origen son cuadriláteros 'degenerados', en que el origen es un nodo duplicado. En las siguientes figuras, se puede ver el mallado inicial el mallado tras 2 refines, (tercermallado), y con este mallado, la solución aproximada y las direcciones de los vectores tensión.



Resultados

Operando con mallado con polares, la evolución en los nodos del mallado inicial, a lo largo de 6 mallas, desde el inicio al quinto refine, es:

Evol (refine=5, tras 6 mallados)=

1	3.45804	3.46794	3.46487	3.46280	3.46196	3.46167
2	3.18788	3.23093	3.24161	3.24430	3.24497	3.24514
3	3.18276	3.22965	3.24131	3.24422	3.24495	3.24513
4	3.17199	3.22712	3.24071	3.24407	3.24491	3.24512
5	3.16267	3.22515	3.24023	3.24395	3.24488	3.24511
6	3.15912	3.22446	3.24005	3.24391	3.24487	3.24511
7	2.53932	2.58168	2.59252	2.59524	2.59593	2.59610
8	2.53263	2.58015	2.59215	2.59515	2.59590	2.59609
9	2.51961	2.57711	2.59140	2.59497	2.59586	2.59608
10	2.50942	2.57463	2.59078	2.59481	2.59582	2.59607
11	2.50579	2.57372	2.59056	2.59476	2.59580	2.59607
12	1.47954	1.50545	1.51216	1.51386	1.51428	1.51439
13	1.47409	1.50421	1.51186	1.51378	1.51426	1.51438
14	1.46384	1.50179	1.51127	1.51363	1.51423	1.51437
15	1.45600	1.49987	1.51079	1.51351	1.51420	1.51437
16	1.45318	1.49917	1.51061	1.51347	1.51419	1.51436
17	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
18	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
19	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
20	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000
21	0.00000	0.00000	0.00000	0.00000	0.00000	0.00000

Se observa la evolución favorable de la convergencia

La comparación con la solución exacta en la función de torsión y el flujo, calculados en los nodos:

```
KontaRefine=1 (2a malla)
Max error Fun Nodos: 2.283662e-002
Max error Flujos Nodos: 1.695247e+000
KontaRefine=3 (4a malla)
Max error Fun Nodos: 1.428476e-003
Max error Flujos Nodos: 4.261209e-001
KontaRefine=5 (6a malla)
Max error Fun Nodos: 1.317042e-004
Max error Flujos Nodos: 1.066387e-001
```

Se observa que los errores en la función de torsión se van reduciendo más al refinar que los del flujo en los nodos. Ello parece natural pues en el flujo se utilizan las derivadas y la función aproximadora por elementos finitos cuadriláteros, continua en el valor de la función de campo, no lo es en el gradiente (derivadas primeras) al calcularse en los bordes entre elementos.

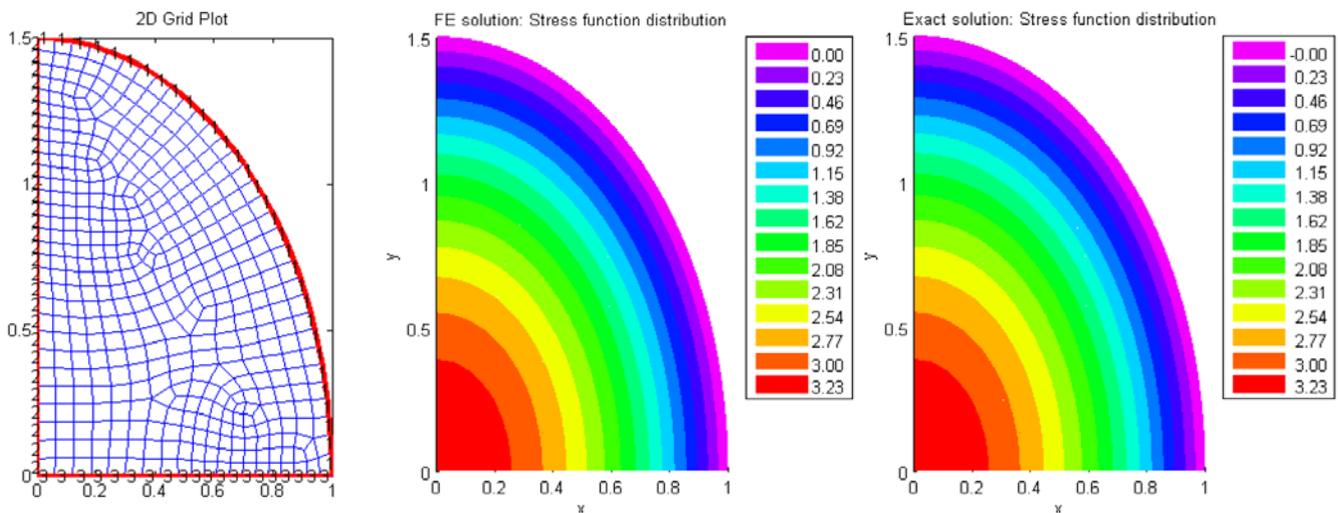
Tiempos de resolución del sistema de ecuaciones de EF, en matriz 'dispersa', con un procesador Intel i7 Q740 1.73 GHz, 8.0 GBytes de memoria física, sistema operativo de 64 bits (2014 julio):

```
KontaRefine = 2 (3a malla)...Elapsed time is 0.421419 seconds
KontaRefine = 3.(4a malla)...Elapsed time is 1.625564 seconds
KontaRefine = 4.(5a malla)...Elapsed time is 6.410178 seconds
KontaRefine = 5.(6a malla)...Elapsed time is 27.325508 seconds
```

El número de elementos finito cuadriláteros en el 6º mallado es de 16384, el número de nodos 16523, incluidos los de borde. La matriz de 'rigidez' global es de 16523x16523 almacenada y procesada con almacenamiento 'disperso' ('sparse'), que ocupa en memoria 2774192 bytes con Matlab, con estructura 'dispersa', en coma flotante doble precisión. El número de elementos no nulos en K es de 36801, con estructura en banda.

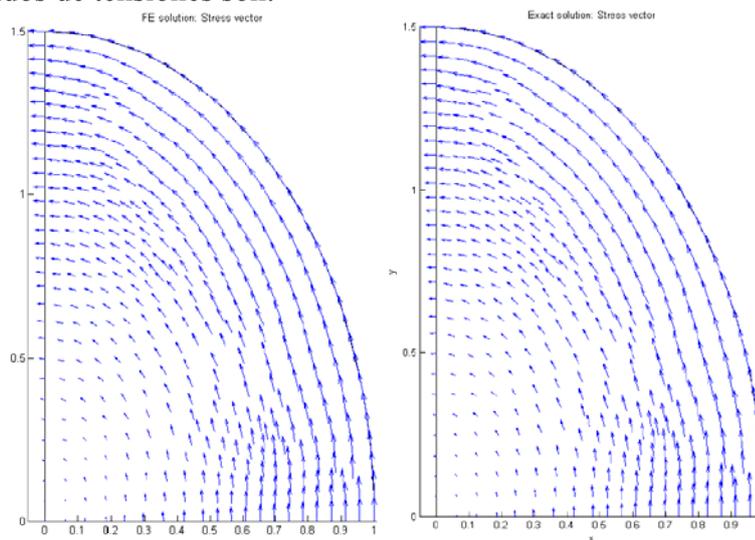
La gestión de evolución de la respuesta en los nodos iniciales se hace en main2D_refine, adaptado para mallado con polares, pues a diferencia del refine del caso con bordes poligonales (se mantiene la numeración de nodos a lo largo de los mallados), en los sucesivos mallados polares esa numeración cambia, aunque los nodos iniciales siguen presentes. Para referenciarlos en al final, se puede trabajar con distancias.

Trabajando con el mallado generado de la figura con ANSYS:



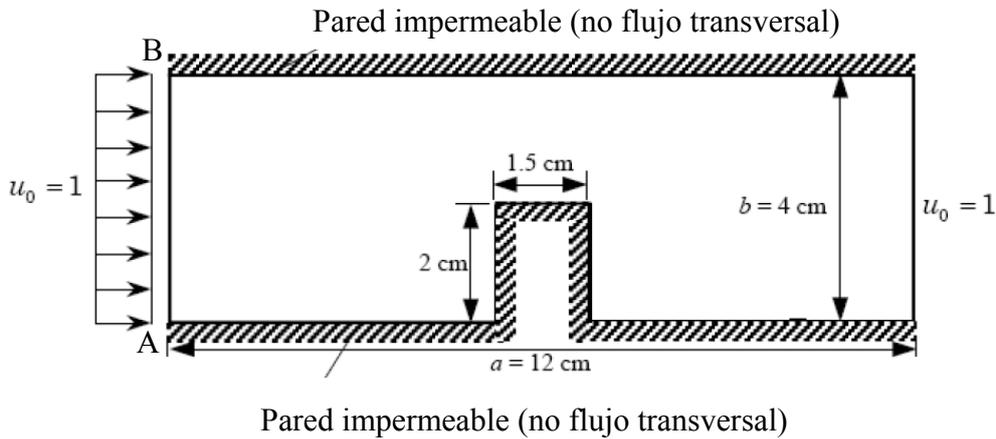
Las condiciones de borde son:

$u=0$ sobre el borde 1 (el 'curvo') y $\partial u/\partial n=0$ sobre los bordes 2 y 3, los lados rectos del dominio reducido por simetría). Los resultados de tensiones son:



Por tanto, la resolución por elementos finitos da soluciones fiables. En el dibujo del vector de tensiones, se observa un movimiento de torsión en la sección elíptica.

Prob PII-3. Flujo confinado en torno a una muro rectangular



El flujo irrotacional de un fluido ideal (es decir, un fluido no viscoso) en torno a un muro (ver la figura anterior) se va a analizar mediante el método de elementos finitos con el programa 2dBVP en Matlab. La ecuación que representa este flujo es:

$$-\nabla^2 u = 0 \quad \text{en } \Omega,$$

donde u puede ser cualquiera de las dos funciones: a) u es función de corriente $\Psi(x,y)$ o b) u es potencial de velocidad $\phi(x,y)$.

a) Si u es la función de corriente Ψ , las componentes de la velocidad del campo de flujo son:

$$u_1 = -\partial\Psi/\partial y \quad , \quad u_2 = +\partial\Psi/\partial x$$

b) Si u es el potencial de velocidad ϕ , las componentes de la velocidad del campo de flujo son:

$$u_1 = -\partial\phi/\partial x \quad , \quad u_2 = -\partial\phi/\partial y$$

En ambos casos, el campo de velocidades no está afectado por un término constante en la solución u. Por simetría, basta modelar la mitad izquierda del dominio. Se elige el punto inferior izquierdo como origen. Primero se puede crear una malla a mano y refinarla, o con un programa como ANSYS y utilizar la función loadFromGridFile.m para leer dicha malla y resolver el problema utilizando las dos formulaciones.

SE PIDE:

- . Calcular la solución en cada elemento y comparar los resultados con las dos formulaciones a) y b).
- . Obtener el dibujo de vectores velocidad y el de curvas de nivel de los potenciales.

Hay que tener cuidado con el tipo de condiciones de contorno que se utilicen. Seguidamente se indican las justificaciones básicas para las condiciones de contorno mostradas en las figuras.

a) Formulación con función de corriente:

Puesto que $u_2 = \partial\Psi/\partial x = 0$ en el borde AB, la función de corriente Ψ sólo depende de y en AB.

Como $u_1 = -\partial\Psi/\partial y = cte = u_0 = 1$ en AB, la función Ψ es lineal en y en AB, esto es $\Psi(x_{AB}, y_{AB}) = c + d \cdot y$. El valor de c se puede tomar arbitrario, no influye en la solución de la EDP.

Recordar que al flujo perpendicular a una línea de corriente es cero. Por tanto, \perp se deben considerar las paredes del dominio como líneas de corriente!! Debido a la simetría respecto de la vertical en medio de la figura, basta con la mitad izquierda del dominio para el análisis. Puesto que el campo de velocidades depende de la diferencia relativa de dos líneas de corriente, se tomará el valor de la línea de corriente que coincide con el punto inferior izquierdo, (digamos el A) con $\Psi_A = 0$, y entonces se determinará el valor de Ψ en la pared superior con la condición de que u_0 es la velocidad del fluido paralela a la línea de corriente. Puesto que u_0 se da en la entrada al dominio, se determinará el valor en la línea de corriente en el punto superior izquierdo (digamos el B) integrando la ecuación anterior del modo siguiente:

$$\int_{\Psi_A}^{\Psi_B} d\Psi = \Psi_B - \Psi_A = \int_{y_A}^{y_B} -u_0 dy = -u_0(y_B - y_A) \quad \text{o bien,}$$

$$\Psi_B = \Psi_A - u_0 \cdot (y_B - y_A)$$

Puesto que la velocidad a la entrada es constante, Ψ varía linealmente en la entrada desde $\Psi_A = 0$ en el punto A, hasta el valor $-u_0 \cdot y_B$ en el punto B. Estas condiciones de contorno se pueden resumir así:

. Sobre el borde izquierdo, se especifica Ψ de valor $\Psi = -y \cdot u_0$, con valor $\Psi = -4 \cdot u_0$, en la pared superior, y $\Psi = 0$ en la pared inferior, que incluye el borde del resalte rectangular.

. Sobre el borde 6, que está en la línea de simetría del dominio, la velocidad vertical es nula, $u_2 = 0$, allí $u_2 = \partial\Psi/\partial x = 0$, luego la proyección del gradiente de Ψ sobre la normal exterior al borde 6:

$$\text{grad}(\Psi) \cdot \mathbf{n} = [\partial\Psi/\partial x = 0, \partial\Psi/\partial y] \cdot [1, 0] = 0, \text{ esta es una condición natural.}$$

b) Formulación con potencial de velocidades:

El hecho de que $u_2 = -\partial\phi/\partial y = 0$ en toda la pared superior y todo el borde inferior del dominio, da las condiciones de contorno en esos bordes.

A lo largo de AB, la velocidad u_0 está dada: $-\text{grad}(\phi) = -(\partial\phi/\partial x, 0) = -(u_0, 0) = (u_0, 0)$, $\text{grad}(\phi) = (-u_0, 0)$.

Al programa 2dBVP se ha de dar $\text{grad}(\phi) \cdot \mathbf{n} = \alpha + \beta \cdot u$, \mathbf{n} normal exterior al recinto, aquí $\mathbf{n} = (-1, 0)$:

$\text{grad}(\phi) \cdot \mathbf{n} = \partial\phi/\partial n = (-u_0, 0) \cdot (-1, 0) = u_0 = \alpha + \beta \cdot u$, como no depende de u será: $\alpha = u_0$, $\beta = 0$

A lo largo de la línea vertical de simetría geométrica (antisimetría del sistema) se ha de saber bien ϕ o bien $\partial\phi/\partial n = \partial\phi/\partial x$. Puesto que $u_1 = -\partial\phi/\partial x$ no se conoce en ese lugar, se supondrá que ϕ es conocido y se hace ϕ igual a una constante (por ejemplo cero) para completar las condiciones de contorno. Haciendo $\phi = 0$ sobre la línea vertical de simetría elimina el movimiento de sólido rígido en la solución. Puesto que el campo de velocidades depende del gradiente de ϕ , cualquier valor constante asignado a ϕ sobre esa línea dará los mismos resultados para $\nabla\phi$. Estas condiciones de contorno se resumen como sigue:

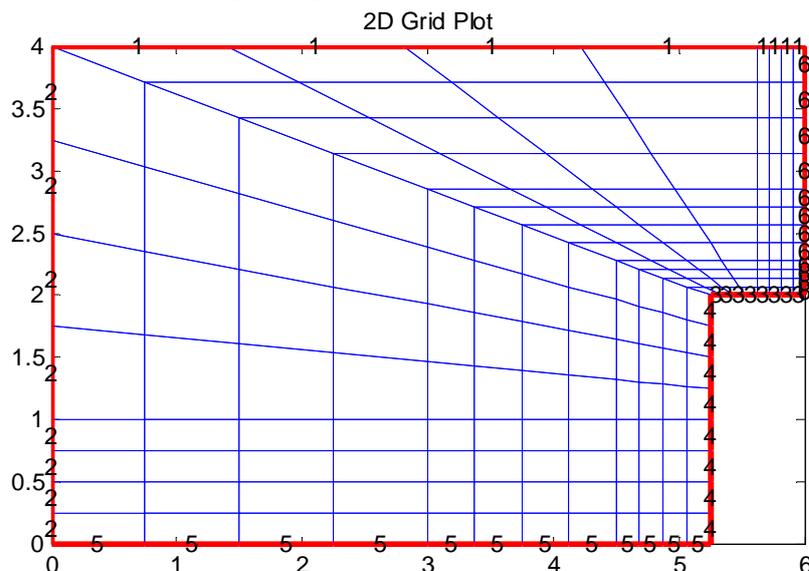
. En el borde derecho, se especifica que $\phi = 0$.

. $\partial\phi/\partial n$ se especifica con valor cero en las paredes superior e inferior impermeables, incluyendo el borde rectangular de resalte.

. Sobre el borde izquierdo, $\partial\phi/\partial n = u_0$

PII-3. Solución

La malla que corresponde a los resultados que se presentan es:



PII-3 a) Resolución con función de corriente:

Las condiciones de contorno son:

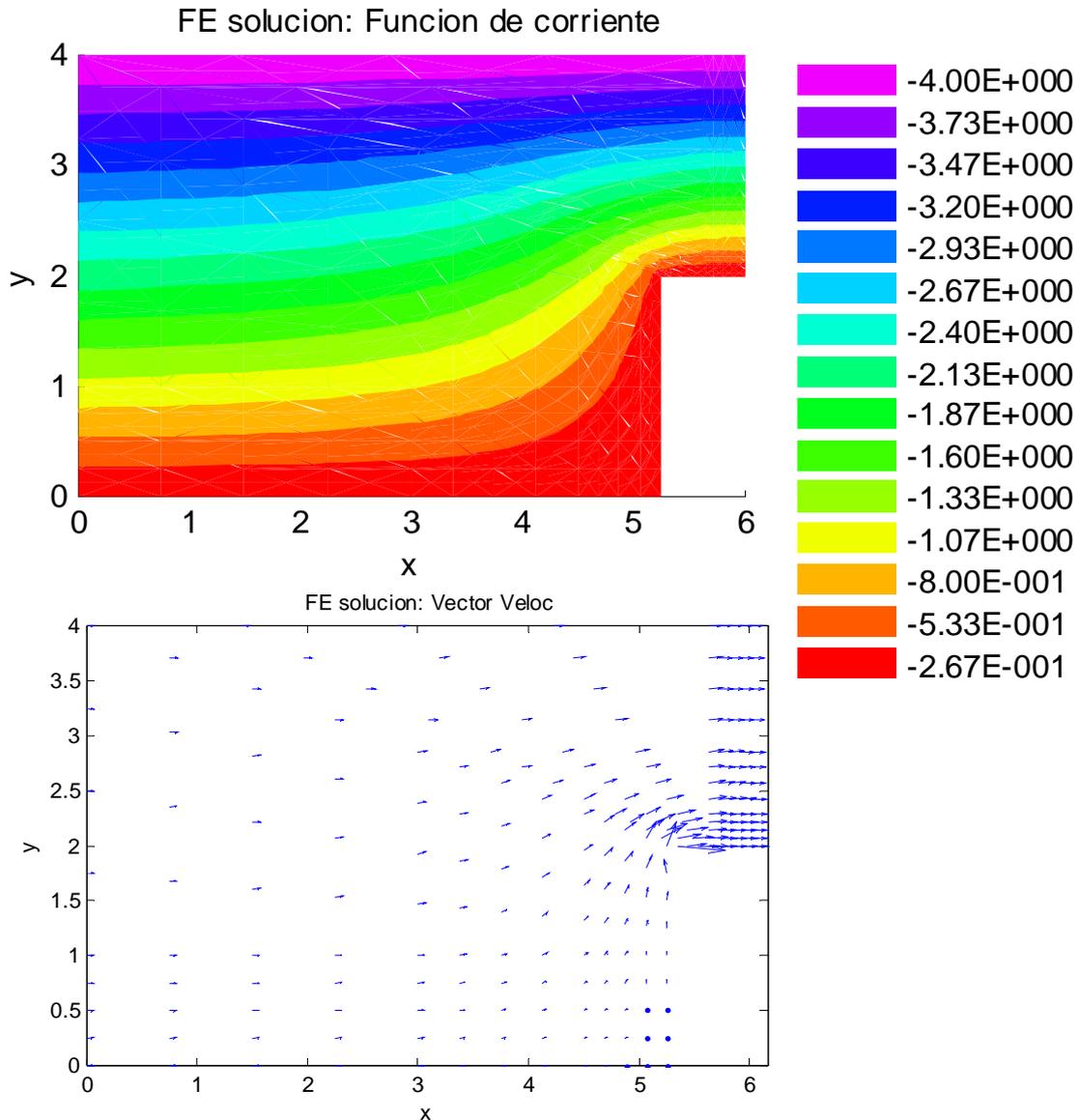
. En el borde izquierdo 2, se especifica Ψ de valor $\Psi = -y \cdot u_0$, $\Psi = -4 \cdot u_0$ sobre el borde superior 1, y $\Psi = 0$ sobre los bordes 3, 4 y 5. Estas son condiciones esenciales.

. Sobre el borde 6, que está en la línea de simetría del dominio, la velocidad vertical es nula, $u_2=0$, allí $\partial\Psi/\partial x=0$, luego la proyección del gradiente de Ψ sobre la normal exterior al borde 6:

$\text{grad}(\Psi) \cdot \mathbf{n} = [\partial\Psi/\partial x=0, \partial\Psi/\partial y] \cdot [1,0] = 0$, esta es una condición natural.

Nota adicional general: Conviene aclarar la definición de las condiciones de contorno. Condición esencial de contorno significa que se especifica el valor sobre el borde de la correspondiente variable incógnita.

Condición ‘natural’ de contorno significa que sólo se especifica la ‘derivada normal’ (por ejemplo el flujo, la velocidad) de la variable incógnita, sin embargo el valor de la variable incógnita (temperatura, velocidad,...), es desconocido.



PII-3 b) Resolución con potencial de velocidades:

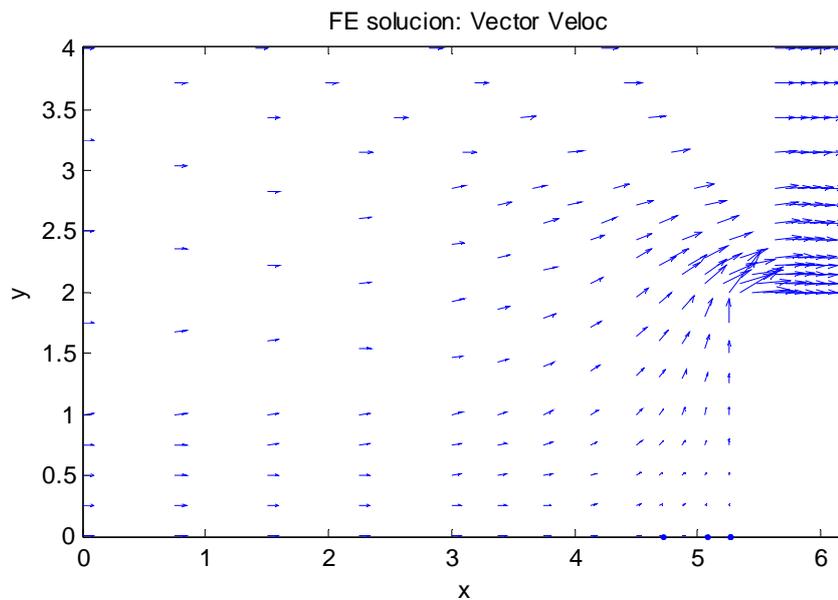
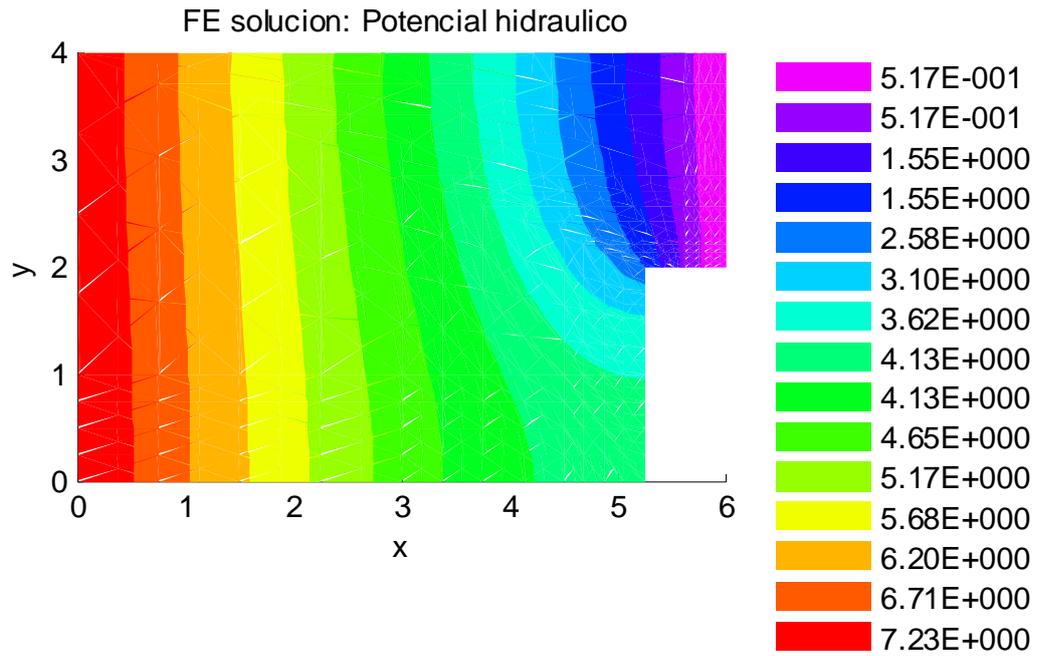
Las condiciones de contorno son:

- . Sobre el borde izquierdo 2, se especifica ϕ mediante $\partial\phi/\partial n=u_0$.
- . Sobre el borde derecho 6, línea de simetría del dominio, $\phi=0$. Esta es condición de borde esencial.
- . Sobre los bordes 1, 3, 4, 5, $\partial\phi/\partial n=0$ condiciones de borde naturales,.

PII.2. Discusión

(1). Se obtiene aproximadamente el mismo campo de velocidades con la función de corriente que con el potencial de velocidades, así se comprueba que ambos métodos se han implementado correctamente.

Además, los vectores velocidad ‘fluyen’ a lo largo del borde del resalte rectangular de acuerdo a la condición de no filtración allí. Sin embargo, hay un vector velocidad en la formulación de función de corriente que ‘penetra’ en el resalte, lo que puede deberse a errores numéricos.



(2) De la teoría del flujo potencial sobrepasando un resalte rígido, hay un punto de ‘estancamiento’ en $(5.25,0)$. Del gráfico de velocidades, se aprecia que la velocidad se anula en ese punto. Además se sabe que la velocidad del flujo alcanza su valor máximo en la parte superior del resalte, lo que es también consistente con la formulación por elementos finitos.

(3) En las líneas de nivel de la función de corriente, se observa que siguen un patrón de líneas paralelas a los bordes inferior y superior del dominio, lo que tiene sentido pues se sabe que el flujo no puede cruzar las líneas de corriente, debe tener un comportamiento similar a los vectores velocidad. Así lo indican los gráficos de velocidades.

(4) De las líneas de nivel de la función potencial, se ve que su comportamiento es perpendicular a las paredes del resalte, lo que debe ocurrir pues la velocidad en sentido perpendicular a las paredes del resalte es nula, y el gradiente del potencial debe ser paralelo a esas paredes.

Todo lo anterior son índices de verificación de la fiabilidad de la solución por elementos finitos.