Informe de la ejecución del deber de Métodos numéricos para EDP
Doctorado Matemáticas Aplicadas
Fausto Fabián Crespo Fernández

Se adaptó el código en Matlab del artículo de Alberty, Carstensen, Funken. Remarks around 50 lines of Matlab:short finite element implementation.

i) Para el problema 1 tomamos la función de la parte derecha $f$ para que la solución exacta sea $u(x,y) = sen(\pi x)\, sen(\pi y)$, el dominio es el cuadrado $[0,1]x[0,1]$ y el sistema EDP es:
$-\Delta u + u = f \; en \; [0,1]x[0,1] = \Omega$
$u = 0 \; en \; \partial\Omega$

O sea:
$(2\pi^2 + 1)sen(\pi x)sen(\pi y) = f \;\; en \; [0,1]x[0,1] = \Omega.$

La formulación débil de este problema: Hallar $u \in H_0^1(\Omega)$
$\int_\Omega \nabla u \, \nabla v \; dx + \int_\Omega \mathrm{uv\, dx} = \int_\Omega \mathrm{fv\, dx} \quad \forall v \in H_0^1(\Omega)$ (I)
porque se toma la función test $v$ que se anule en la frontera de $\Omega$ (al igual que la solución).

Y basta que se cumpla esta ecuación para la base del espacio vectorial $\{V_h\}$ que aproxima a $H_0^1(\Omega)$ :
$\int_\Omega \nabla u \, \nabla \eta_j \; dx + \int_\Omega u\eta_j \; \mathrm{dx} = \int_\Omega \mathrm{f}\eta_i \; \mathrm{dx}.$ (Ia)
y como $u = \sum x_k \, \eta_k$
$\int_\Omega \nabla \eta_i \, \nabla \eta_j \; dx + \int_\Omega \eta_i \, \eta_j \; \mathrm{dx} = \int_\Omega \mathrm{f}\eta_i \; \mathrm{dx}.$ (II)

Los espacios $\{V_h\}$ se escogen globalmente continuos y afines en cada elemento triangular que forma parte de la descomposición del dominio. Las integrales en (II) se aproximan como:
$\sum_{T\epsilon\mathcal{T}} \int_T \nabla \eta_i \, \nabla \eta_j \; dx + \sum_{T\in\mathcal{T}} \int_T \eta_i \, \eta_j \; \mathrm{dx} = \sum_{T\in\mathcal{T}} \int_T \mathrm{f}\eta_i \; \mathrm{dx}$ (III)

y las funciones techo se escogen de forma tal que $\eta_j(x_k, y_k) = \delta_{jk}, j, k = 1, \ldots, N$

La matriz de stiffness o rigidez y la matriz de masa corresponden al primer término y segundo término de la izquierda, respectivamente.

En una discretización usando triángulos tales que para el triangulo $T$ con vértices $(x_1, y_1)$, $(x_2, y_2)$ y $(x_3, y_3)$ y funciones base de $V_h$: $\eta_1$, $\eta_2$ y $\eta_3$, tenemos que $\eta_j(x_k, y_k) = \det\begin{pmatrix} 1 & x & y \\ 1 & x_{j+1} & y_{j+1} \\ 1 & x_{j+2} & y_{j+2} \end{pmatrix} / \det\begin{pmatrix} 1 & x_j & y_j \\ 1 & x_{j+1} & y_{j+1} \\ 1 & x_{j+2} & y_{j+2} \end{pmatrix}$

Por lo que $\nabla \eta_i(x,y) = \frac{1}{2|T|}\begin{pmatrix} y_{j+1} - y_{j+2} \\ x_{j+2} - x_{j+1} \end{pmatrix}$

Y $|T|$ es el área de $T$ y la suma de los índices es suma módulo 3.
Y la matriz de stiffness es

$$A_{jk} = \int_{\mathrm{T}} \nabla\eta_i \, \nabla{\eta_k}^T \, dx = \frac{|T|}{(2|T|\ )^2}(y_{j+1} - y_{j+2}, x_{j+2} - x_{j+1})\begin{pmatrix} y_{j+1} - y_{j+2} \\ x_{j+2} - x_{j+1} \end{pmatrix}$$
$$= \frac{|T|}{2} G G^T$$

con $G = \begin{pmatrix} 1 & x & y \\ 1 & x_{j+1} & y_{j+1} \\ 1 & x_{j+2} & y_{j+2} \end{pmatrix}^{-1} \begin{pmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}$

Para la matriz de masa:

$$B_{jk} = \sum_{T \in \mathcal{T}} \int_T \ \eta_j \, \eta_k \, \mathrm{dx}$$

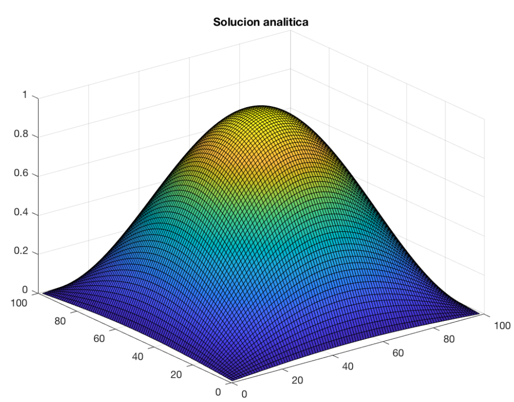
y $\int_T \ \eta_i \, \eta_k \, \mathrm{dx} = \frac{1}{24} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix}$

La parte derecha se aproxima como:

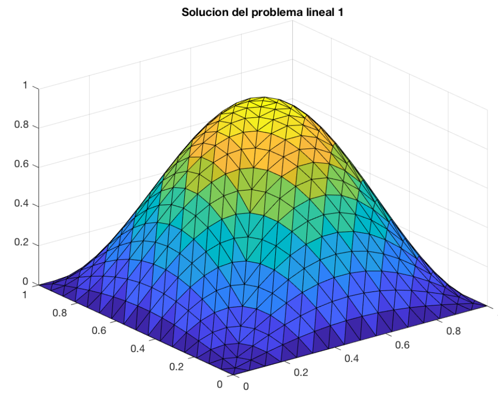
$$b_i = \int_T \mathrm{f}\eta_i \, \mathrm{dx} \approx \frac{1}{6} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} f(x_S, y_S\ )$$

donde $(x_S, y_S\ ) = (\frac{x_1+x_2+x_3}{3}, \frac{y_1+y_2+y_3}{3})$ es el centro de masa del triangulo $T$.
El problema se reduce a encontrar los coeficientes $x$ de la solución expresada como combinación lineal en la base:
$(A + B)x = b$
o sea $x = (A + B)^{-1} b$
y la solución es $u = \sum_{k=1}^{N} x_k \, \eta_k$

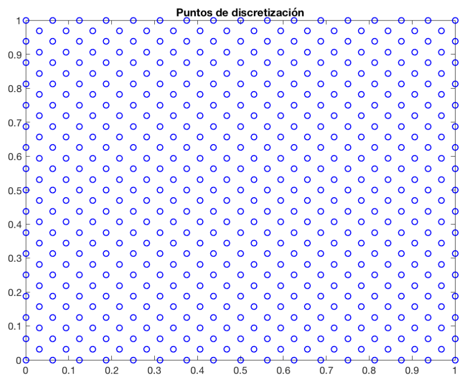
La solución analítica exacta es : $u(x, y) = sen(\pi x)\ sen(\pi y)$

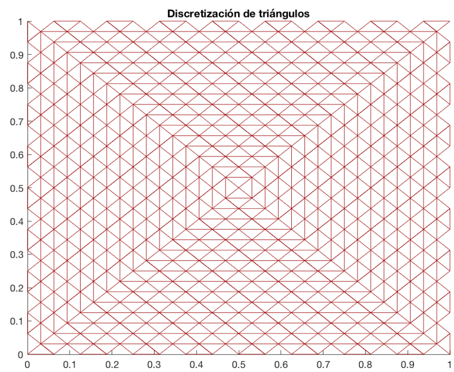


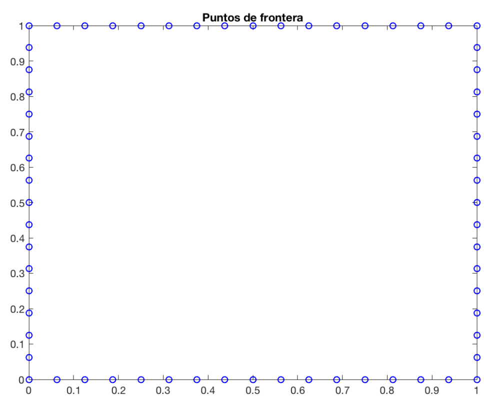

Y la solución aproximada con la discretización FEM es




Se usó una discretización de 1024 triángulos del dominio $[0,1]x[0,1]$.





El estimador del error en la norma $L_2$:

$\|e_h\|^2{}_{L^2} = \|u - u_h\|^2{}_{L^2} = a(\varphi,\ e_h) = a(\varphi - \varphi_h,\ e_h)$ (V)
(pues $a(\varphi_h,\ e_h) = 0\ \forall \varphi_h \in V_h$)

donde $a(.,\ .)$ es la forma bilineal y $\varphi$ es la solución del problema auxiliar adjunto

$$-\Delta\varphi = e_h \text{ en } \Omega$$
$$\varphi = 0 \quad \text{ en } \partial\Omega$$

En general un estimador a priori para el problema variacional $a(u,v) = F(v)\ \forall v \in V$, con $u_h$ la aproximación por FEM de grado $r > 0$, y la solución $u \in H^{p+1}(\Omega)$ con $p > 0$ dado, entonces
$\|u - u_h\|^2{}_{L^2} \leq Ch^{s+1}|u|_{H^{s+1}}$ con $s = \min\{r, p\}$.
($r$ es el grado de los polinomios que se usan en las funciones techo y $h = \max_{T\in\mathcal{T}} h_T$ y $h_T = \max_{x,y\in T}|x - y|$ que en este caso es $h = \frac{1}{16} = 0.0625$.

Este estimación es difícil de calcular por que la constante $C$ no es conocida a priori.

Pero podemos usar la ecuación (V) para estimar el error como $\|u - u_h\|^2{}_{L^2} = (u_h - u)^T B(u_h - u)$ y
$B$ la matriz de masa, lo que resulta en 3.4292e-07

El número de condición de la matriz de rigidez (stiffness) $A$ fue de 11.0415 y el de la matriz de masa $B$ de 302.2243 y el número de condición de $A + B$ es de 286.5179

Para la norma $H_1$tenemos si $u \in H^{r+1}(\Omega)$ se tiene que
$\|u - u_h\|^2{}_{H^1(\Omega)} \leq Ch^s|u|_{H^{s+1}(\Omega)}$ con $s = \min\{r, p\}$.
Donde $|u|_{H^{s+1}(\Omega)}$ es la seminorma en $H^{s+1}(\Omega)$

El error en la norma $H_1$: $(u_h - u)^T A(u_h - u) =$ 9.8417e-05
También se puede usar $((u_h - u)^T(A + B)(u_h - u) =$ 9.8760e-05

i) Para el problema 2 tomamos la función de la parte derecha $f$ para que la solución exacta fuera $u(x,y) = sen(2\pi x)\, cos(2\pi y)$, el dominio era el cuadrado $[0,1]x[0,1]$ y el sistema:
$-\Delta u + u = f\ en\ [0,1]x[0,1] = \Omega$
$u = u_d\ en\ \partial\Omega$

O sea:
$(8\pi^2 + 1)sen(2\pi x)cos(2\pi y) = f$.
En este caso la formulación débil es exactamente igual al caso anterior pues no tenemos condiciones de Neuman y al aplicar el teorema de Green nos queda exactamente los mismo.

Para este caso donde tenemos condiciones de frontera Dirichlet no homogéneas, hacemos el cambio $z = u - u_d$ de forma tal que $z \equiv 0$ en la frontera de $\Omega$. Así entonces $u = z + u_d$ y $\nabla u = \nabla z + \nabla u_d$, por lo que nos queda:
De
$\int_\Omega \nabla u\, \nabla v\, dx + \int_\Omega \mathrm{uv\, dx} = \int_\Omega \mathrm{fv\, dx} \quad \forall v \in H_0^1(\Omega)$
que

$\int_\Omega \nabla z\, \nabla v\, dx + \int_\Omega \nabla u_d\, \nabla v\, dx + \int_\Omega \mathrm{zv\,dx} + \int_\Omega u_d \mathrm{v\,dx} = \int_\Omega \mathrm{fv\,dx} \qquad \forall v \in H_0^1(\Omega)$

$\int_\Omega \nabla z\, \nabla v\, dx + \int_\Omega \mathrm{zv\,dx} = \int_\Omega \mathrm{fv\,dx} - \int_\Omega u_d \mathrm{v\,dx} - \int_\Omega \nabla u_d\, \nabla v\, dx \qquad \forall v \in H_0^1(\Omega)$

Que se debe cumplir en especial para la base de los espacios y que se aproxima como una suma por todos los triángulos:

$\sum_{T\epsilon\mathcal{T}} \int_T \nabla z\, \nabla \eta_j\, dx + \sum_{T\in\mathcal{T}} \int_T z\, \eta_j\, \mathrm{dx} = \sum_{T\epsilon\mathcal{T}} \int_T \mathrm{f}\eta_i\, \mathrm{dx} - \sum_{T\in\mathcal{T}} \int_T u_d \eta_i\, \mathrm{dx} - \sum_{T\in\mathcal{T}} \int_{\mathrm{T}} \nabla u_d\, \nabla v\, dx$

Y teniendo en cuenta que $u_d = \sum_{k=1}^{n} u_k\, \eta_k$ y haciendo $u = \sum_{k=1}^{N} x_k\, \eta_k$, la parte derecha se aproxima como una suma por todos los triángulos:

$$b_i = \sum_{T\in\mathcal{T}} \int_T \mathrm{f}\eta_i\, \mathrm{dx} - \sum_{k=1}^{n} u_k \sum_{T\epsilon\mathcal{T}} \int_T \eta_i\, \eta_k\, dx - \sum_{k=1}^{n} u_k \sum_{T\epsilon\mathcal{T}} \int_T \nabla\eta_i\, \nabla\eta_k dx$$

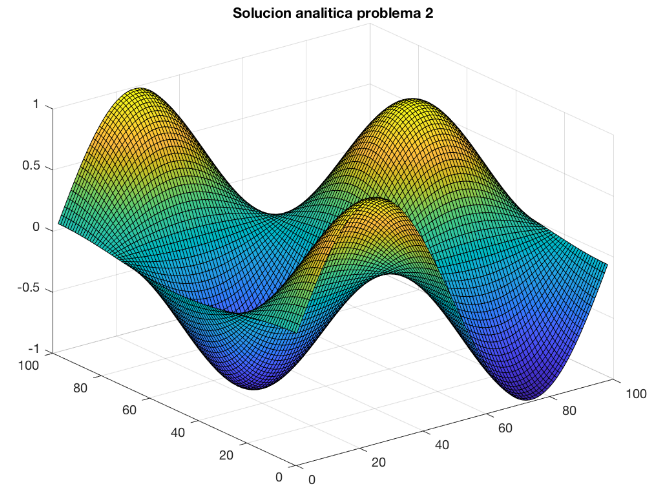
$$\approx \sum_{T\in\mathcal{T}} \frac{1}{6} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} f(x_S, y_S) - \sum_{k=1}^{n} u_k \sum_{T\in\mathcal{T}} \frac{1}{24} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} \begin{pmatrix} 2 & 1 & 1 \\ 1 & 2 & 1 \\ 1 & 1 & 2 \end{pmatrix} - \sum_{k=1}^{n} u_k \sum_{T\epsilon\mathcal{T}} \int_T \nabla\eta_i\, \nabla\eta_k\, dx$$

$$= \sum_{T\in\mathcal{T}} \frac{1}{6} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} f(x_S, y_S) - \left(B_{i,:}{}^T u\right) - \left(A_{i,:}{}^T u\right)$$

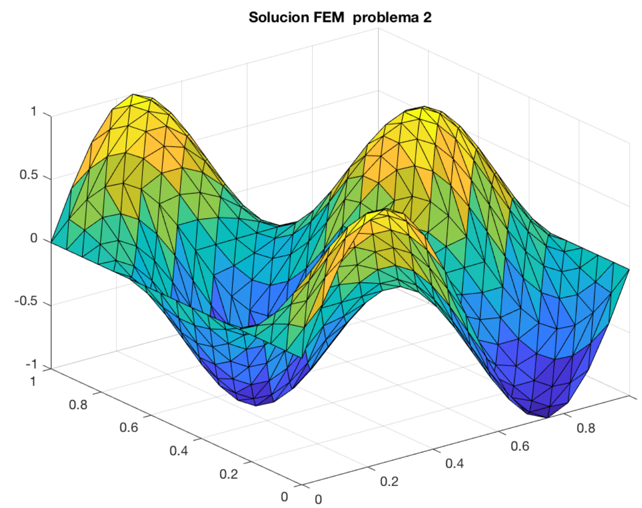
donde $(x_S, y_S) = (\frac{x_1+x_2+x_3}{3}, \frac{y_1+y_2+y_3}{3})$ es el centro de masa del triangulo $T$.
Y $u$ es el vector de coeficientes $u_k$ de la función $u_d$ en la base $\{\eta_k\}$

Por tanto $b = \sum_{T\epsilon\mathcal{T}} \frac{1}{6} \det\begin{pmatrix} x_2 - x_1 & x_3 - x_1 \\ y_2 - y_1 & y_3 - y_1 \end{pmatrix} f(x_S, y_S) - (A + b)u$
Y el sistema se reduce a $(A + B)x = b$ y $x = (A + B)^{-1} b$
Y la solución final es: $u = u_d + \sum_{k=1}^{N} x_k\, \eta_k$

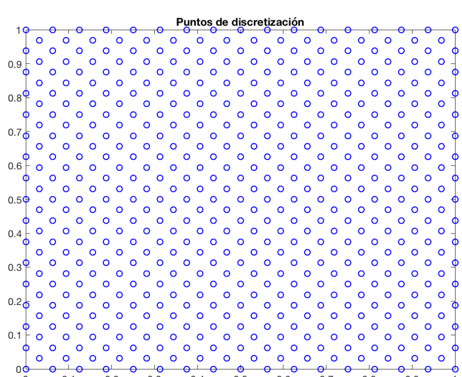
La solución analítica exacta es : $u(x, y) = sen(2\pi x) cos(2\pi y)$ en $[0,1]x[0,1] = \Omega$

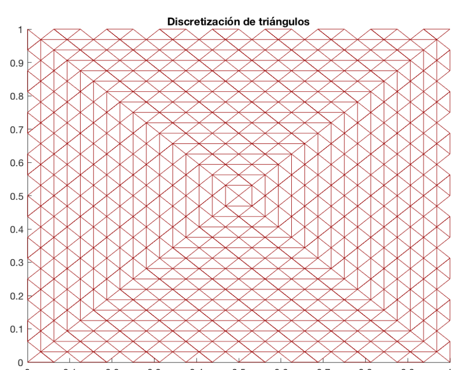


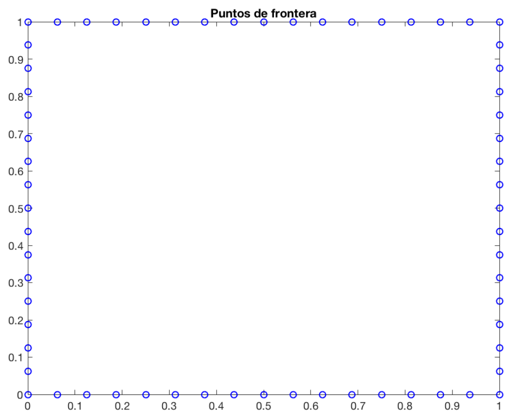

Y la solución aproximada con la discretización FEM es

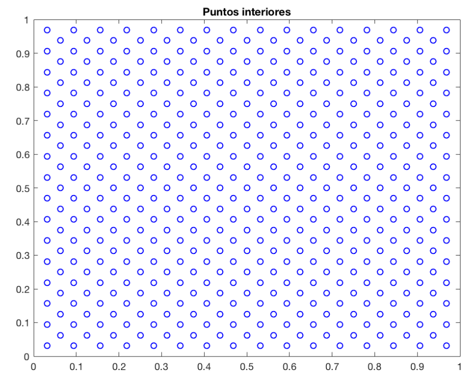



Se usó una discretización de 1024 triángulos del dominio $[0,1]x[0,1]$.






Análogamente al problema 1, el error con la norma $L_2$ fue: 7.6768e-06

El error en la norma $H_1$: 0.0014
Y usando $((u_h - u)^T (A+B)(u_h - u) = 0.0014$

Bibliografía


1- Alberty, Carstensen, Funken. Remarks around 50 lines of Matlab:short finite element implementation.

2- Fuzheng Gao, Lin Mu. 2014. ON L$^2$ ERROR ESTIMATE FOR WEAK GALERKIN FINITE ELEMENT METHODS FOR PARABOLIC PROBLEMS. Journal of Computational Mathematics.

3- Notas curso.


Apéndice

Código en Matlab:

```
load('coordenadas.mat')

load('triangulos.mat')

load('frontera.mat')

%%%elementos y triangulos

figure
plot(C(:,1), C(:,2),'bo')
title('Puntos de discretizaciÛn')
```

```
figure
for j= 1:size(E,1)
    hold on
    plot(C(E(j,:),1),C(E(j,:),2),'Color',[.6 0 0])
    hold on

end
title('DiscretizaciÛn de tri·ngulos')

pointsFrontier=C(unique(F),:)
figure
plot(pointsFrontier(:,1), pointsFrontier(:,2),'bo')
title('Puntos de frontera')


FreeNodes= setdiff(1: size(C,1), unique(F))
interiorPoints=C(FreeNodes,:)
figure
plot(interiorPoints(:,1), interiorPoints(:,2),'bo')
title('Puntos interiores')
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

MatrizMasa=sparse(size(C,1),size(C,1));
A = sparse(size(C,1),size(C,1));
%partes derechas
f1 = sparse(size(C,1),1);
f2 = sparse(size(C,1),1);

%matriz rigidez o stiffness
for j=1:size(E,1)
   A(E(j,:),E(j,:))=A(E(j,:),E(j,:))+stima3(C(E(j,:),:));
end

%matriz masa
for j=1:size(E,1)
   MatrizMasa(E(j,:),E(j,:))=MatrizMasa(E(j,:),E(j,:))+det([1,1,1;
C(E(j,:),:)'])*[2,1,1;1,2,1;1,1,2]/24;
end

r=det([1,1,1; C(E(1,:),:)']);
z1= inline('sin(pi*x)*sin(pi*y)*(2*pi^2+1)','x','y');
z2=inline('sin(2*pi*x)*cos(2*pi*y)*(8*pi^2+1)','x','y');
g=inline('sin(2*pi*x)*cos(2*pi*y)','x','y');
for j = 1:size(E,1)
 v=sum(C(E(j,:),:))/3;
 f1(E(j,:)) = f1(E(j,:))+r*z1(v(1),v(2))/6;
 f2(E(j,:)) = f2(E(j,:))+r*z2(v(1),v(2))/6 ;
end

A(unique(F),:)=sparse(length(A(unique(F),unique(F))),size(C,1));
A(unique(F),unique(F))=speye(length(A(unique(F),unique(F))));

f1(unique(F))=sparse(length(f1(unique(F))),1);

f2(unique(F))=sparse(length(f2(unique(F))),1);
```

```
pointsFrontier=C(unique(F),:);

u= sparse(size(C,1),1);
u(unique(F))=u_d(C(unique(F),:));
f2=f2-(A+MatrizMasa)*u;

u2(FreeNodes)=(A(FreeNodes, FreeNodes)+MatrizMasa(FreeNodes,
FreeNodes))\f2(FreeNodes);
%u2(FreeNodes)= u2(FreeNodes)+u_d(C(FreeNodes,:));
u2(unique(F))=u_d(C(unique(F),:));



u1=(A+MatrizMasa)\f1;
figure
trisurf(E,C(:,1),C(:,2),full(u1'));
title('Solucion FEM  problema 1');

 figure
 trisurf(E,C(:,1),C(:,2),full(u2'));
 title('Solucion FEM  problema 2');
%problema 1
ut1=@(x,y)(sin(pi.*x).*sin(pi.*y));

%problema 2
ut2=@(x,y)(sin(2*pi.*x).*cos(2*pi.*y));
 h=0.01;
 [x1,y1]=meshgrid(h:h:1-h,h:h:1-h);
 UF=ut1(C(:,1),C(:,2));
 Ufe=ut1(x1,y1);
 figure
 surf(Ufe);
title('Solucion analitica problema 1');
display('Para el problema 1');
display('Con L2');
 errorL2=(u1-UF)'*MatrizMasa*(u1-UF);
 display(errorL2);
 display('N˙mero de condiciÛn de matriz de  masa');
display(condest(MatrizMasa));
display('Con H1');
errorH1=(u1-UF)'*A*(u1-UF);
display(errorH1);
error=(u1-UF)'*(A+MatrizMasa)*(u1-UF);
 display('N˙mero de condiciÛn de matriz de  rigidez');
display(condest(A));
display('N˙mero de condiciÛn de suma de matrices');
display(condest(MatrizMasa+A));


UG=ut2(C(:,1),C(:,2));
 Uge=ut2(x1,y1);
figure
 surf(Uge);
title('Solucion analitica problema 2');

display('Para el problema 2');
```

```
display('Con L2');
errorL2=(u2'-UG)'*(MatrizMasa)*(u2'-UG);
 display(errorL2);
 display('N˙mero de condicion de matriz de masa');
display(condest(MatrizMasa));
display('Con H1');
errorH1=(u2'-UG)'*A*(u2'-UG);
 display(errorH1);
 error=(u2'-UG)'*(A+MatrizMasa)*(u2'-UG);
 display(error);
 display('N˙mero de condicion de matriz de  rigidez');
display(condest(A));
display('N˙mero de condicion de suma de matrices');
display(condest(MatrizMasa+A));
```

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

En el archivo aparte u_d.m tenemos

```
function value = u_d ( u )

   n = size ( u, 1 );

  value(1:n) = sin (2.0 * pi * u(1:n,1) ) .* cos( 2.0 * pi * u(1:n,2) );

  return
end
```