MAP431 - Projet

BIAD Anas / EL KHADIR Bachir

May 14, 2014

Question 1

Soit $v \in H_0^1(\Omega)$
On multiplie l'équation par la fonction test v et on intègre par parties en utilisant la formule de Green :

$$\begin{aligned}\int_\Omega f(x)v(x)dx &= \int_\Omega v(x)u_\mu(x)dx - \int_\Omega \frac{d}{dx}(\kappa_\mu(x)\frac{du_\mu}{dx}(x))v(x)dx \\ &= \int_\Omega v(x)u_\mu(x)dx + \int_\Omega \kappa_\mu(x)\frac{dv}{dx}(x)\frac{du_\mu}{dx}(x)dx\end{aligned}$$

Donc la forumation variationnelle du problème (3) s'écrit :

$$a_\mu(u_\mu, v) = b(v)$$

où :

$$a_\mu(u_\mu, v) = \int_\Omega v(x)u_\mu(x)dx + \int_\Omega \kappa_\mu(x)\frac{dv}{dx}(x)\frac{du_\mu}{dx}(x)dx$$

$$b(v) = \int_\Omega f(x)v(x)dx$$

Question 2

Les coefficients de A:

$$A_{i,j} = a_\mu(\varphi_i, \varphi_j) = \begin{cases} \frac{2}{h^2}\int_{K_i}\kappa_\mu(x)dx + \frac{2}{3}h & \text{si } i = j \\ -\frac{1}{h^2}\int_{K_i}\kappa_\mu(x)dx + \frac{h}{4} & \text{si } j = i+1 \\ -\frac{1}{h^2}\int_{K_{i-1}}\kappa_\mu(x)dx + \frac{h}{4} & \text{si } j = i-1 \\ 0 & \text{sinon} \end{cases}$$

Les coefficients de B:

$$\begin{aligned}B_i = b(\varphi_i) &= \int_0^1 \varphi_i(x)sin(2\pi x)dx \\ &= \int_0^1 \varphi(\frac{x-x_i}{h})sin(2\pi x)dx \\ &= h\,sin(2\pi x_i)\,(sinc(\pi h))^2\end{aligned}$$

Question 3

Le fichier calcul.sce contenant les fonctions calculant A, B puis la solution X

```
clear
N = 100
h = 1/(N+1)
omega_2 = [0.19, 0.21; 0.38, 0.42; 0.58, 0.62; 0.79, 0.81]
mailles = [1:N]*h

// Calcul de l'intégrale
```

```
function y=int_k(a, b, mu)
    if a > b then
        [b, a] = (a, b)
    end
    k = 1
    for i=omega_2'
        if i(1) <= a & i(2) >= b then
            k = mu
            break
        end
    end
    y = k * (b-a)
endfunction

// Calcul du coefficient de A
function y=coeff_a(i, j, mu)
    y = 0
    if i == j then
         y = 2/3 * h +  2/(h**2) * int_k(i*h, (i+1)*h, mu)
    end
    if abs(i-j) == 1 then
        y = h / 4 - 1/(h**2) * int_k(i*h, j*h, mu)
    end

endfunction

// Calcul de la matrice A
function A=calc_A(mu)
    A = zeros(N, N)
    for i=1:N
        for j=1:N
            A(i, j) = coeff_a(i, j, mu)
        end
    end
endfunction

// Calcul de la matrice B
function B=calc_B()
    B = zeros(N)
    for i=1:N
        B(i) = h  * sin(2*%pi*i*h) * (sinc(%pi*h))**2
    end
endfunction

// Calcul de la solution
function X=resoudre(mu)
    A = calc_A(mu)
    B = calc_B()
    X = inv(A) * B
endfunction
```

Le fichier question3.sce qui calcule et affiche le résultat pour $\mu \in \{0.01, 0.1, 1\}$

```
exec('calcul.sce',-1)

xset("color",6)
a=gca();
a.font_size=5;
```

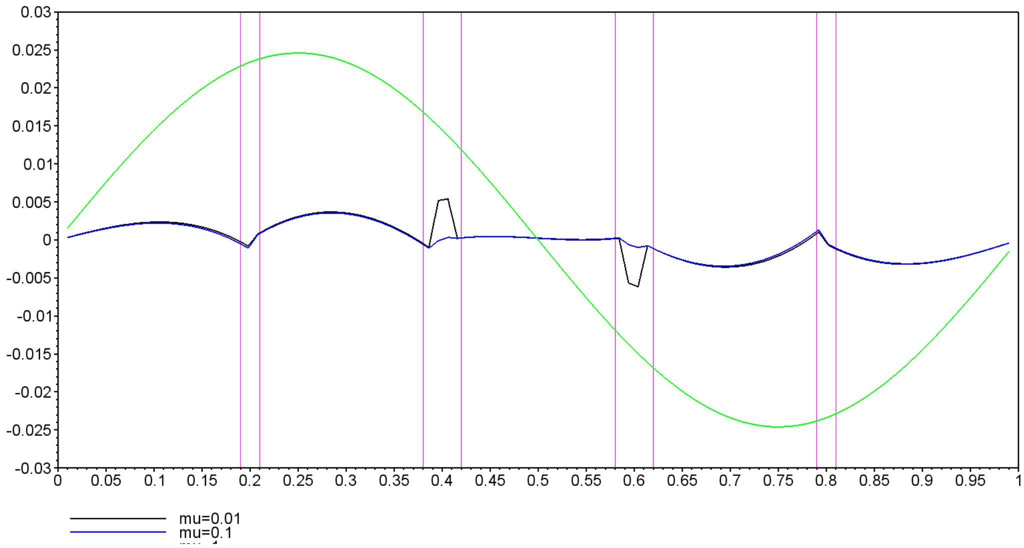

Figure 1: courbe de u_μ pour $\mu \in \{0.01, 0.1, 1\}$

```
for i=omega_2'
    rect = [i(1) 0.03 (i(2)-i(1)) 0.06]
    xrect(rect)
end

xset("color",1)
a.thickness = 2;
mu = [0.01, 0.1, 1]
colors = ['r', 'g', 'b']
solutions = zeros(N, 3)
for i=1:3
    A = calc_A(mu(i))
    B = calc_B()
    solutions(:, i) = inv(A) * B
end
rect = [0, -0.03, 1, 0.03]
leg = "mu=0.01@mu=0.1@mu=1"
plot2d(mailles, solutions, 1:3, "111",leg, rect)
```

Le résultat graphique

Commentaire

commenatire ici

Question 4

En injectant la relation:

$$u_\mu^{RB} = \sum_{j=1}^{N_0} X_{\mu,j}^{RB} u_j$$

Dans:

$$a_\mu(u_\mu^{RB}, u_i) = b(u_i)$$

et On trouve :

$$\sum_{j=1}^{N_0} X_{\mu,j}^{RB} a_\mu(u_j, u_i) = b(u_i)$$

Ainsi X_μ^{RB} est solution du système linéaire :

$$A_\mu^{RB} X_\mu^{RB} = B^{RB}$$

Avec :

$$(A_\mu^{RB})_{i,j} = a_\mu(u_j, u_i)$$

et :

$$(B^{RB})_i = b(u_i)$$

On a :

- a_μ est symétrique
- a_μ est positive car $(\forall v \in H_1^0(\Omega))\ a_\mu(v,v) = \int_\Omega (v(x))^2 dx + \int_\Omega \kappa_\mu(x)(\frac{dv}{dx}(x))^2 dx >= 0$
- Si $a_\mu(v,v) = 0$ alors $\int_\Omega (v(x))^2 dx = 0$, donc $v = 0$. Ainsi a_μ est définie

A_μ^{RB} est la matrice de a_μ dans la base des u_i, donc A_μ^{RB} est symétrique , définie positive.

Question 5

on a :

$$\begin{aligned}
(A_\mu)_{i,j} &= a_\mu(u_\mu, v) \\
&= \int_\Omega \varphi_i(x)\varphi_j(x)dx + \int_{\Omega_1} \frac{d\varphi_i}{dx}(x)\frac{d\varphi_j}{dx}(x)dx + \mu \int_{\Omega_2} \frac{d\varphi_i}{dx}(x)\frac{d\varphi_j}{dx}(x)dx \\
&= \int_\Omega \varphi_j(x)\varphi_i(x)dx + \int_{\Omega_1} \frac{d\varphi_j}{dx}(x)\frac{d\varphi_i}{dx}(x)dx + \int_{\Omega_2} \frac{d\varphi_j}{dx}(x)\frac{d\varphi_i}{dx}(x)dx \\
&= (A_0)_{i,j} + (A_1)_{i,j}
\end{aligned}$$

Où l'on a posé:

$$A_0 = \left(\int_\Omega \varphi_j(x)\varphi_i(x)dx + \int_{\Omega_1} \frac{d\varphi_j}{dx}(x)\frac{d\varphi_i}{dx}(x)dx \right)_{i,j}$$

$$A_1 = \left(\int_{\Omega_2} \frac{d\varphi_j}{dx}(x)\frac{d\varphi_i}{dx}(x)dx \right)_{i,j}$$

Et donc on a:

$$A_\mu = A_0 + \mu A_1$$

D'autre part , on a

$$\begin{aligned}
(A_\mu^{RB})_{i,j} &= a_\mu(u_j, u_i) \\
&= [u_i]^T_{(\varphi_1,\ldots,\varphi_N)} A_\mu \, [u_j]_{(\varphi_1,..,\varphi_N)} \\
&= X_i^T A_\mu X_j \\
&= X_i^T A_0 X_j + \mu X_i^T A_1 X_j \\
&= (A_0^{RB})_{i,j} + \mu (A_1^{RB})_{i,j}
\end{aligned}$$

et :

$$(B^{RB})_i = b(u_i) = B^T X_i$$

Question 6

```
exec('calcul.sce',-1)

N0 = 3
tab_mu = [0.05, 0.2, 1]

// Calcul de A0, A1 et B
A0 = calc_A(0)
A1 = calc_A(1) - A0
```

```
B = calc_B()

// Famille des solutions Xi
solutions = zeros(N, N0)
for j=1:N0
    solutions(:, j) = resoudre(tab_mu(j))
end

// Evaluation de A0_RB, A1_RB et B_RB
A0_RB = zeros(N0, N0)
A1_RB = zeros(N0, N0)
for i=1:N0
    for j=1:i
        a_rb = solutions(:,i)' * (A0 * solutions(:,j))
        A0_RB(i, j) = a_rb
        A0_RB(j, i)  = a_rb

        a_rb = solutions(:,i)' *  (A1 * solutions(:,j))
        A1_RB(i, j) = a_rb
        A1_RB(j, i)  = a_rb
    end
end

B_RB = zeros(N0)
for i=1:N0
    B_RB(i) = B' * solutions(:,i)
end

// Resolution du petit système linéaire
function U=resoudre2(mu)
    U = solutions * ( inv(A0_RB + mu * A1_RB) * B_RB )
endfunction

exec('calcul2.sce',-1)

mu = 0.075
u_rb = resoudre2(mu)
u_mu = resoudre(mu)


plot(mailles, u_rb, 'r')
plot(mailles, u_mu, 'g')

plot(mailles, abs(u_mu-u_rb), 'b')
```

Graphie ici Commentaire ici

Question 7

```
exec('calcul2.sce',-1)

//Méthode de Monte Carlo

function Y = aleatoire()
    //Y = 0.99 * rand() + 0.01
    Y = 10**(rand()*2-2)
endfunction


M = 1000
```

```
E_MC = zeros(N)
for i=1:M
    mu = aleatoire()
    E_MC = E_MC + resoudre2(mu)
end

E_MC = E_MC/M

U_mu_bar = resoudre2(0.99/log(100))

plot(mailles, E_MC, "r")
plot(mailles, U_mu_bar)
```

Graphie ici Commentaire ici

Question 8

```
// Définition de Omega
border B1(t=0,1) {x=t;y=0 ;label =1;};
border B2(t=0,1) {x=1;y=t ;label =1;};
border B3(t=1, 0) {x=t;y=1 ;label =1;};
border B4(t=1,0) {x=0;y=t ;label =1;};

real Rc = 0.075;
border C11(t=0,2*pi){x=1*0.2+Rc*cos(t); y=1*0.2+Rc*sin(t); label=2;};
border C12(t=0,2*pi){x=1*0.2+Rc*cos(t); y=2*0.2+Rc*sin(t); label=2;};
border C13(t=0,2*pi){x=1*0.2+Rc*cos(t); y=3*0.2+Rc*sin(t); label=2;};
border C14(t=0,2*pi){x=1*0.2+Rc*cos(t); y=4*0.2+Rc*sin(t); label=2;};
border C21(t=0,2*pi){x=2*0.2+Rc*cos(t); y=1*0.2+Rc*sin(t); label=2;};
border C22(t=0,2*pi){x=2*0.2+Rc*cos(t); y=2*0.2+Rc*sin(t); label=2;};
border C23(t=0,2*pi){x=2*0.2+Rc*cos(t); y=3*0.2+Rc*sin(t); label=2;};
border C24(t=0,2*pi){x=2*0.2+Rc*cos(t); y=4*0.2+Rc*sin(t); label=2;};
border C31(t=0,2*pi){x=3*0.2+Rc*cos(t); y=1*0.2+Rc*sin(t); label=2;};
border C32(t=0,2*pi){x=3*0.2+Rc*cos(t); y=2*0.2+Rc*sin(t); label=2;};
border C33(t=0,2*pi){x=3*0.2+Rc*cos(t); y=3*0.2+Rc*sin(t); label=2;};
border C34(t=0,2*pi){x=3*0.2+Rc*cos(t); y=4*0.2+Rc*sin(t); label=2;};
border C41(t=0,2*pi){x=4*0.2+Rc*cos(t); y=1*0.2+Rc*sin(t); label=2;};
border C42(t=0,2*pi){x=4*0.2+Rc*cos(t); y=2*0.2+Rc*sin(t); label=2;};
border C43(t=0,2*pi){x=4*0.2+Rc*cos(t); y=3*0.2+Rc*sin(t); label=2;};
border C44(t=0,2*pi){x=4*0.2+Rc*cos(t); y=4*0.2+Rc*sin(t); label=2;};


int Nb=50;
int Nc=20;

mesh Th=buildmesh(B1(Nb)+B2(Nb)+B3(Nb)+B4(Nb)+C11(Nc)+C12(Nc)+C13(Nc)+C14(Nc)+C21(Nc)+C22(Nc)+C23(Nc)+C24(Nc)

// Espaces d'éléments finis
fespace Vh(Th, P1);
fespace Xh(Th, P0);

// Calcul de A0 et A1
Xh coeffB = (region==16);
Xh coeffC = (region<16);

varf A0p(u,v,solver=UMFPACK) =
int2d (Th)( u*v + coeffB * (dx(u)*dx(v)+dy(u)*dy(v)) ) + on(1,u=0);
matrix A0 = A0p(Vh,Vh);

varf A1p(u,v,solver=UMFPACK) =
```

```
int2d (Th)( coeffC * (dx(u)*dx(v)+dy(u)*dy(v)) ) + on(1,u=0);
matrix A1 = A1p(Vh,Vh);

// Choix de P0 et initialisation des matrices réduites A0red et A1red
int nrb=3;
real [int] select(nrb);
select[0]=.1;select[1]=.5;select[2]=1.;
real [int,int] A0red(nrb,nrb);
real [int,int] A1red(nrb,nrb);
real[int] Bred(nrb);

// Calcul du deuxième terme
func f = sin(2*pi*x)*sin(2*pi*y);
varf L(u, v)=int2d(Th) (v*f) + on(1, u=0);
real[int] F = L(0, Vh);

//Calcul des fonctions de la base réduite
Vh [int] solutions(nrb);
Vh u=0;

for(int k=0;k<nrb;k++){
        real mu = select(k);
        matrix A = A0+mu*A1;
        set(A,eps=1e-10);
        u[]=A^-1*F;
        solutions[k]=u;
}

int nddl= Vh.ndof;

real [int] buffer(nddl);
for(int i=0;i<nrb;i++){
        for(int j=0;j<nrb;j++){
                buffer = (A0*solutions[j][]);
                A0red(i,j) =solutions[i][]'*buffer;
                buffer = (A1*solutions[j][]);
                A1red(i,j) =solutions[i][]'*buffer;
        }
        Bred(i) = F' * solutions[i][];
}

// Résolution du sysètme pour mu=0.075
for(int i=0; i < 1000; i++) {
        real mu = 0.01*i;
        matrix mA0 = A0red;
        matrix mA1 = A1red;
        matrix mA = (mA0 + mA1*mu);
        set(mA,eps=1e-10);
        real[int] buff =  (mA^-1) * Bred;
        Vh U=0;

        for(int j=0;j<nrb;j++)
                U[] = U[] +  (solutions[j][] * buff(j));
        plot(U, wait=1);
}
```