M1 | Mathématiques Fondamentales et Appliquées (MFA)
Université Paris–Sud | D4MA10 : MAO Option calcul scientifique
Centre d'Orsay | Année 2008 – 2009

Solution TP3 : Exemple de mise en oeuvre de la mthode d'lments finis

Ce TP a pour but de suivre les tapes de mise en oeuvre de la mthode des lments finis sur un problme modle monodimensionnel. Les lments finis concerns seront les lments finis Lagrange de degr 1 et 2.

Exercice - 1

Problme rsoudre :

$$\left| \begin{array}{ll} -\Delta u = f & \text{dans } \Omega, \\ u = 0 & \text{sur } \partial\Omega, \end{array} \right. \qquad \text{o} \qquad \Omega =]0,1[, f \in L^2(\Omega) \tag{1}$$

Q-1 **: Formulation Variationnelle.** En appliquant le dmarche dcrite dans le cours, on aboutit au problme variationnel suivant

$$\boxed{\text{Chercher } u \in H_0^1(\Omega) \text{ tel que } \quad \int_\Omega u'(x)v'(x)\,dx = \int_\Omega f(x)v_h(x)\,dx \quad \forall v \in H_0^1(\Omega)}$$

La dmonstration de l'existence et l'unicit de la solution de ce problme est laisse ici en exercice.

Q-2 **: Discrtisation par lments finis P_r-Lagrange ($r = 1, 2$).**

On commence par introduire un maillage de Ω, c'est–dire une suite de N_{so} points

$$0 = x_1 < \ldots < x_{N_{so}-1} < x_{N_{so}} = 1 \quad \text{ce qui gnre } N_{ma} = N_{so} - 1 \text{ mailles } I_i = [x_i, x_{i+1}].$$

On introduit l'espace lments finis P_r-Lagrange, $X_h^r = \left\{v_h \in \mathcal{C}^0([0,1]) \text{ tel que } u_h|_{[x_i,x_{i+1}]} \in \mathbb{P}_r\right\}$.
C'est bien un sous-espace de $H^1(\Omega)$. On introduit alors le sous-espace de X_h^r prenant en compte les conditions aux limites : $X_{0h}^r = \{v_h \in X_h^r \text{ tel que } v_h(0) = v_h(1) = 0\}$.

Q-3 **:** X_{0h}^1 est un sous-espace de $H_0^1(\Omega)$ de dimension $N_{so} - 2$, correspondant au nombre de sommets internes. Et X_{0h}^2 est un sous-espace de $H_0^1(\Omega)$ de dimension $N_{ma} + N_{so} - 2$ correspondant au total des degrs de libert situs aux sommets des mailles (soit $N_{so} - 2$) plus ceux situs au centre des mailles (soit N_{ma}).
Et le problme discret s'crit :

$$\boxed{\forall v_h \in X_{0h}^r, \quad \int_\Omega u_h'(x)v_h'(x)\,dx = \int_\Omega f(x)v_h(x)\,dx}.$$

Q-4 **: Formulation matricielle.** Si on peut expliciter une base $(\varphi_i)_{1\le i\le \text{dim}(X_{0h}^r)}$ de X_{0h}^r, alors en posant $u_h = \sum_{i=1}^{\text{dim}(X_{0h}^r)} u_i\varphi_i$ et $U = (u_1, \ldots, u_{\text{dim}(X_{0h}^r)})^t$ le problme discret ci-dessus se ramne la rsolution du systme linaire : $AU = b$, avec $A_{i,j} = \int_\Omega \varphi_i'(x)\varphi_j'(x)\,dx \quad 1 \le i,j \le \text{dim}(X_{0h}^r)$,
$b_i = \int_\Omega f(x)\varphi_i(x)\,dx \quad 1 \le i \le \text{dim}(X_{0h}^r)$.

Seulement part dans le cas particulier de la dimension une, il n'est pas ais dans le cas gnral d'exhiber une base de X_{0h}^r et de calculer les coefficients de A et b partir de cette base. Nanmoins, on sait identifier une base locale de X_h^r (c'est dire une base sur chaque lment), ce titre, on calcule la matrice A et le vecteur b

directement dans X_h^r, par assemblage des matrices et vecteurs locaux. Puis on introduit les conditions aux limites aprs assemblage. **Les exercices 2 et 3 dcrivent cette procdure. La comprhension de ces exercices assure la comprhension de la mise en oeuvre de la mthode des lments finis dans un cadre gnral.**

Exercice - 2 *lments finis P1-Lagrange*

On dsigne par `X` les sommets x_i du maillage qu'on numrote de 1 N_{so}.

1. **Table des degrs de liberts.** La position des degrs de libert sont les sommets du maillage. Sur une maille numro i : $I_i = [x_i, x_{i+1}]$, Le sommet 1 est x_i son numro global est i ; le sommet 2 est x_{i+1}, son numro global est $i+1$. Donc le sommet numro $j = 1, 2$ de la maille i a pour numro $i + j - 1$. D'o la table de degr de libert

$$l2g(i,j) = i + j - 1 \quad 1 \leq i \leq N_{ma}, 1 \leq j \leq 2.$$

2. **Traitement lmentaire**. Soit $I_i = [x_i, x_{i+1}]$, une maille quelconque.
   (a) Explicitation des fonctions de base locale $P1-$Lagrange

$$\phi_1(x) = -\frac{x - x_{i+1}}{x_{i+1} - x_i}, \quad \phi_2(x) = \frac{x - x_i}{x_{i+1} - x_i}$$

   On pose $h_i = x_{i+1} - x_i$ le diamtre de la maille I_i.

   (b) Calcul de la matrice lmentaire A^i et du vecteur lmentaire b^i

$$A^i = \int_{x_i}^{x_{i+1}} \begin{bmatrix} \phi_1{}' \\ \phi_2{}' \end{bmatrix} \begin{bmatrix} \phi_1{}' & \phi_2{}' \end{bmatrix} dx = \frac{1}{h_i} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}$$

$$b^i = \int_{x_i}^{x_{i+1}} f(x) \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} dx = \frac{h_i}{2} \begin{bmatrix} f(x_i) \\ f(x_{i+1}) \end{bmatrix}$$

   Pour calculer le second membre, on a utilis la formule des trapzes :
   $\int_{x_i}^{x_{i+1}} \psi(x)\, dx = \frac{x_{i+1}-x_i}{2} (\psi(x_i) + \psi(x_{i+1}))$.

   (c) Script `Matlab` de construction de A^i, b^i

```
function [Ai,bi]=elementlaplaceP1(xi1,xi2, i)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
hi=xi2 - xi1;
Ai = (1/hi)*[1,-1;-1,1];
bi=(hi/2)* [ f(xi1); f(xi2)];
```

3. **Assemblage** : Pour assembler la matrice A et le second membre b, on boucle sur les lments et pour chaque lment, on construit sa matrice locale et sont vecteur local et on les accumulent dans A et b.

script Matlab d'assemblage pour P1

```
function [A,b] = assembleLaplaceP1(x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% function [A,b] = assembleLaplaceP1(x)
%%
%% fonction qui assemble la matrice du laplacien P1 en 1D
%% ENTREE: x les sommets du maillage
%% SORTIE: A la matrice
%%         b le vecteur second membre
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nso = length(x); % nombre de sommets
Nma = Nso -1 ;   % nombre de mailles
% table des degres de liberte
for i = 1:Nma
    l2g(i,1) = i;
```

```
    l2g(i,2) = i+1;
end
%allocation d'espace
A=zeros(Nso,Nso);
b=zeros(Nso,1);
%boucle d'assemblage
for i=1:Nma
    %On recupere les sommets de l'element
    x1 = x(i); x2 = x(i+1);
    % On calcule la matrice et le second membre elementaires
    [Ai,bi] = elementlaplaceP1(x1,x2,i);
    %Et on assemble la matrice et le vecteur elementaires dans les matrice et vecteur globaux
    A(l2g(i,1:2), l2g(i,1:2)) = A(l2g(i,1:2), l2g(i,1:2)) + Ai(1:2,1:2);
    b(l2g(i,1:2)) = b(l2g(i,1:2)) + bi(1:2);
end
```

4. **Conditions aux limites** :
   Pour introduire les conditions aux limites $u_h(0) = u_h(1) = 0$, on commence par localiser les degrs de libert situs en $x = 0$ et $x = 1$, par leur numro global et leurs coordonnes : pour $x = 0$, on a $idl = 1, xdl = 0$ et pour $x = 1$, on a $idl = N_{so}$ et $xdl = x_{Nso}$.
   On dispose alors aprs de plusieurs approches dont deux sont les suivantes :
   - Par pnalisation : on choisit une trs grande valeur tgv, et pour chaque indice idl du degr de libert devant porter la condition de Dirichlet $u_h(xdl) = g(xdl)$, on pose $A(idl, idl) = tgv, \quad b(idl) = g(xdl) * tgv$.
   - Par terme unit sur la diagonale : On cre le vecteur U_D nul de mme taille que U. Puis pour chaque idl on pose $U_D(idl) = g(xdl)$. On remplace ensuite b par $b = b - AU_D$. Et pour chaque indice idl, on pose $b(idl) = U_D(idl)$ et on met tous les termes des lignes et colonnes idl de A zro, sauf le terme diagonal qu'on met 1.

script Matlab d introduction des conditons auxlimites pour P1

```
function [A,b]=conditionLimiteLaplaceP1(x,A,b)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% function [A,b] = conditionLimiteLaplaceP1(x,A,b)
%%
%% fonction qui introduit les conditions aux limites pour le pb du Laplacien P1 en 1D
%% ENTREE: x les noeuds du maillage
%          A la matrice assemblee
%          b le vecteur assemble
%          entree implicite: g la fonction de la condition de Dirichliet
%% SORTIE: A la matrice A modifiee
%%         b le vecteur b modofie
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nso = length(x); % nombre de sommets
%Nma = Nso -1 ;   % nombre de maille

%CONDITION AUX LIMITES PAR PENALISATION
% tgv = 1e+30;
% A(1,1) = tgv; b(1) = tgv * g(x(1)); % g est la condition de dirichlet
% A(Nso,Nso) = tgv; b(Nso) = tgv * g(x(Nso));

%CONDITION AUX LIMITES PAR MODIFICATION
ud=zeros(Nso,1);ud(1)=g(x(1));ud(Nso)=g(x(Nso));
b=b-A*ud;
b(1) = ud(1);b(Nso)=ud(Nso);
A(:,1) =0;A(1,:) =0;A(:,Nso) =0;A(Nso,:) =0;
A(1,1) =1;A(Nso,Nso) =1;
```

On peut alors crer la fonction `function [A,b] = laplaceP1(X)` qui assemble la matrice et le second membre de la discrtisation par lments finis $P1-$Lagrange du Laplacien.

```
function [A,b]=laplaceP1(x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [A,b]=laplaceP1(x)
% fonction qui assemble la matrice du laplacien P1
% et introduit les conditions aux limites
```

```
% Entree: x maillage
%
% Sortie: A matrice
%         b vecteur second membre global
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[A,b] = assembleLaplaceP1(x);
[A,b] = conditionLimiteLaplaceP1(x,A,b);
```

Exercice - 3 *lments finis P2-Lagrange*

On continue dsigner par `X` les sommets x_i du maillage, qu'il ne faut pas confondre avec la position des degrs de libert.

1. **Table des degrs de liberts.**
   Au niveau global, on choisit de numroter d'abord les degrs de liberts situs sur les sommets (ceci donne N_{so}) et ensuite ceux situs au milieu des mailles. Ainsi sur une maille $I_i = [x_i, x_{i+1}]$, le numro du degr de libert situ en son centre est $N_{so} + i$ c'est–dire somme du numro de la maille et du nombre total des sommets. Les numros globaux des degrs de libert locaux sont donc $(i, N_{so} + i, i + 1)$. D'o la table
   $$l2g(i,1) = i,\ l2g(i,2) = N_{so} + i,\ l2g(i,3) = i + 1, \quad 1 \leq i \leq N_{ma}.$$
   Ce choix permet de numroter en dernier les degrs de libert associs aux noeuds internes aux mailles. D'autres choix sont possibles comme par exemple, numroter suivant l'ordre croissant des abscisses des positions des degrs de libert.
2. **Traitement lmentaire**. Soit $I_i = [x_i, x_{i+1}]$, une maille quelconque et $x_{i+\frac{1}{2}}$ son point milieu.
   (a) Explicitation des fonctions de base locale $P2-$Lagrange
   $$\phi_1(x) = \frac{(x - x_{i+\frac{1}{2}})(x - x_{i+1})}{(x_i - x_{i+\frac{1}{2}})(x_i - x_{i+1})},$$
   $$\phi_2(x) = \frac{(x - x_i)(x - x_{i+1})}{(x_{i+\frac{1}{2}} - x_i)(x_{i+\frac{1}{2}} - x_{i+1})},$$
   $$\phi_3(x) = \frac{(x - x_i)(x - x_{i+\frac{1}{2}})}{(x_{i+1} - x_i)(x_{i+1} - x_{i+\frac{1}{2}})}.$$
   Attention l'ordre (dite numrotation locale) des fonctions de base locale, impos par la table des degrs de libert $l2g$. En effet, la fonction de base locale n^o1 est associe x_i, celle n^o2 est associe $x_{i+\frac{1}{2}}$, et celle n^o3 x_{i+1}.
   (b) Calcul de la matrice lmentaire A^i et du vecteur lmentaire b^i
   On a jug utile d'utiliser les formules de quadrature pour calculer la fois A^i et b^i. Cela nous a incit programmer les fonctions de base de sorte retourner pour chacune d'elles sa valeur et sa drive en un point (donn en argument). (voir ci-dessous)
   (c) script `Matlab` de construction de A^i, b^i (voir ci-dessous)

Fonctions de base locale P2

```
function [v,dv] = baselocalP2(x1,x2,i,x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [v,dv] = baselocalP2(x1,x2,i,x)
% fonction de base P1-Lagrange
% entrï¿½e: [x1,x2] ->intervalle
%   i ->numero local de la fonction de base
%   x ->point d'evaluation
% sortie :
% v -> valeur de la fonction de base
% dv-> valeur de derivee de la fonction de base
%
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009
%  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
xm =(x1+x2)/2;
if(i==1)
      d = (x1-xm)*(x1-x2);
      v = (x-xm)*(x-x2)/d;
     dv = (2*x - xm - x2)/d;
elseif(i==2)
      d = (xm-x1)*(xm-x2);
      v = (x-x1)*(x-x2)/d;
      dv= (2*x - x1 - x2)/d;
elseif(i==3)
       d = (x2-xm)*(x2-x1);
       v = (x-xm)*(x-x1)/d;
      dv = (2*x - xm - x1)/d;
end
```

Calcul lmentaire pour P2

```
function [Ai,bi]=elementlaplaceP2(x1,x2, i)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% matrice et le second membre elementaires
% du Laplacien par EF P2-Lagrange
% Entrees:[x1,x2] intervalle
%        i numero de l'element
%        f fonction second membre fournie dans f.m
% Sortie : Ai matrice 3 x 3
%           bi vecteur 3 x 1
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%formule de quadrature de simpson
w = [1   ,          4,   1]*(x2-x1)/6; % poids
z = [x1 , (x1+x2)/2, x2];           % points
%initialisation
Ai=zeros(3,3);bi=zeros(3,1);
%on boucle sur les points de quadrature
for q=1:3
    wq=w(q);
    zq = z(q);
    %boucle sur les lignes
    for l=1:3
        [vl,dvl] = baselocalP2(x1,x2,l,zq);
        bi(l)= bi(l) + wq * f(zq)*vl;
        %boucle sur les colonnes
        for m=1:3
            [vm,dvm] = baselocalP2(x1,x2,m,zq);
            Ai(l,m)=Ai(l,m) + wq * dvl * dvm;
        end
    end
end
```

3. **Assemblage** : L'assemblage est identique celui prcdent.

   Nous avons modifi le script pour retourner aussi la position des degrs de libert, afin de faciliter la reprsentation graphique de la solution P2.

script Matlab d'assemblage pour P2

```
function [A,b,y] = assembleLaplaceP2(x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [A,b,y] = assembleLaplaceP2(x)
%
% fonction qui assemble la matrice du laplacien P2 en 1D
% ENTREE: x les sommets du maillage
% SORTIE: A la matrice
%         b le vecteur second membre
%         y position des degres de liberte
%
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nso = length(x); % nombre de sommets
Nma = Nso -1 ;   % nombre de mailles
Ndl=Nso + Nma;
% table des degres de liberte
% et leur position
y=[];l2g=[];
for i = 1:Nma
    l2g(i,1) = i;
    l2g(i,2) = i+Nso;
    l2g(i,3) = i+1;
    y(l2g(i,1))=x(i);
    y(l2g(i,3)) = x(i+1);
    y(l2g(i,2)) = (x(i)+x(i+1))/2;
end
%allocation d'espace
A=zeros(Ndl,Ndl);
b=zeros(Ndl,1);
%boucle d'assemblage
for i=1:Nma
    %On recupere les sommets de l'element
    x1 = x(i); x2 = x(i+1);
    % On calcule la matrice et le second membre elementaires
    [Ai,bi] = elementlaplaceP2(x1,x2,i);
    % Et on assemble la matrice et le vecteur elementaires
    A(l2g(i,1:3), l2g(i,1:3)) = A(l2g(i,1:3), l2g(i,1:3)) + Ai(1:3,1:3);
    b(l2g(i,1:3)) = b(l2g(i,1:3)) + bi(1:3);
```

```
end
```

4. **Conditions aux limites** : La prise en compte des conditions aux limites $u_h(0) = u_h(1) = 0$, se droule comme expliqu plus haut. Il faut juste faire attention ici la numrotation des degrs de libert, qui fait que celui situ en x=1, n'est pas le dernier degr de libert, son numro est N_{so} et non $N_{so} + N_{ma}$.

script Matlab d'introduction des conditions aux limites pour P2

```
function [A,b]=conditionLimiteLaplaceP2(x,A,b)
Nso = length(x); % nombre de sommets
Nma = Nso -1 ;   % nombre de maille
Ndl=Nso + Nma;

% Attention le numero du degre de liberte dont la position la position est
% x(Nso) n'est pas Ndl mais plutot Nso

%CONDITION AUX LIMITES PAR PENALISATION
% tgv = 1e+30;
% A(1,1) = tgv; b(1) = tgv * g(x(1)); % g est la condition de dirichlet
% A(Nso,Nso) = tgv; b(Nso) = tgv * g(x(Nso));

%CONDITION AUX LIMITES PAR MODIFICATION
ud=zeros(Ndl,1);
ud(1)=g(x(1));ud(Nso)=g(x(Nso));
b=b-A*ud;b(1) = ud(1);b(Nso)=ud(Nso);
A(:,1) =0;A(1,:) =0;A(:,Nso) =0;A(Nso,:) =0;
A(1,1) =1;A(Nso,Nso) =1;
```

On peut alors creer la fonction `function [A,b] = laplaceP2(X)` qui assemble la matrice et le second membre de la discrtisation par lments finis $P2-$Lagrange du Laplacien.

```
function [A,b,y]=laplaceP2(x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [A,b]=laplaceP2(x)
% fonction qui assemble la matrice du laplacien P2
% et introduit les conditions aux limites
% Entree: x maillage
%
% Sortie: A matrice
%         b vecteur second membre global
%  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[A,b,y] = assembleLaplaceP2(x);
[A,b] = conditionLimiteLaplaceP2(x,A,b);
```

Exercice - 4 *Une comparaison*

On s'intresse prsent au problme suivant

$$\left| \begin{array}{ll} -\epsilon u'' + u' = 1 & \text{dans }]0,1[, \\ u(0) = u(1) = 0 & . \end{array} \right. \tag{2}$$

La solution exacte de cette quation aux drives ordinaires est

$$u(x) = x - \frac{\exp(\frac{x}{\epsilon}) - 1}{\exp(\frac{1}{\epsilon}) - 1}.$$

On gnralise l'tude prcdente pour obtenir un nouveau problme. On cherche une solution $u \in H_0^1(0,1)$ du problme

$$\forall v \in H_0^1(0,1), \quad \epsilon \int_0^1 u'v' + \int_0^1 u'v = \int_0^1 v.$$

On utilise la mthode des lments finis $P1$ puis $P2$ pour trouver une solution approche de ce problme.

Q-1 **:** Le problme discret est ici

$$\boxed{\text{Chercher } u_h \in X_{0h}^r \text{ tel que } \quad \epsilon \int_\Omega u_h'(x) v_h'(x)\,dx + \int_\Omega u_h'(x) v_h(x)\,dx = \int_\Omega f(x) v_h(x)\,dx \quad \forall v_h \in X_{0h}^r,}$$

Avec $r = 1$ pour la discrtisation $P1$ et $r = 2$ pour la discrtisation $P2$, avec $f(x) = 1$ la fonction second membre.

Q-2 **: Constructions pour P1-Lagrange**

1. On commence par traitement lmentaire. Soit I_i une maille ; on construit sa matrice lmentaire et son vecteur lmentaire : $A_{l,m}^i = \epsilon \int_{I_i} (\phi_l^i)'(\phi_m^i)'\,dx + \int_{I_i} (\phi_l^i)(\phi_m^i)'\,dx \quad 1 \le l,m \le 2$

   La matrice locale associe au terme $\int_{I_i} (\phi_l^i)'(\phi_m^i)'\,dx$ est identique celle de l'exercice prcdent, celle associe au terme $\int_{I_i} (\phi_l^i)(\phi_m^i)'\,dx$ est

$$\int_{x_i}^{x_{i+1}} \begin{bmatrix} \phi_1 \\ \phi_2 \end{bmatrix} \begin{bmatrix} \phi_1{}' & \phi_2{}' \end{bmatrix} dx = \frac{1}{2} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}.$$

   D'o

$$A^i = \frac{\epsilon}{h_i} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} + \frac{1}{2} \begin{bmatrix} -1 & 1 \\ -1 & 1 \end{bmatrix}.$$

   Le calcul de b^i reste inchang. On a finalement le script

```
function [Ai,bi]=elementconvdiffP1(xi1,xi2,epsi, i)
hi=xi2 - xi1;
Ai = (epsi/hi)*[1,-1;-1,1] + (1/2)*[-1,1;-1,1];
bi=(hi/2)* [ f(xi1); f(xi2)];
```

2. Puis on crit un script qui assemble les matrices et vecteurs lmentaires et introduit les conditions aux limites

```
function [A,b]=convdiffP1(epsi,x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [A,b] =convdiffP1(x)
% fonction qui assemble la matrice du probleme de convection diffusion
% pour les polynomes de degre 1 et introduit les conditions aux limites
% ENTREE: x -> les noeuds du maillage
%         eps -> le coefficient epsilon
% SORTIE: A -> matrice avec conditions aux limites incorporees
%         b -> vecteur second membre avec conditions aux limites
%         incorporees
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nso = length(x); % nombre de sommets
Nma = Nso -1 ;   % nombre de maille
% table des degre de liberte
A=zeros(Nso,Nso);
b=zeros(Nso,1);
for i = 1:Nma
    l2g(i,1) = i;
    l2g(i,2) = i+1;
end
for i=1:Nma
    %On recupere les sommets de l'element
    x1 = x(i); x2 = x(i+1);
    % On calcule la matrice et le second membre elementaire
    [Ai,bi] = elementconvdiffP1(x1,x2,epsi,i);
```

```
    % et on assemble la matrice et le vecteur elementaires
    A(l2g(i,1:2), l2g(i,1:2)) = A(l2g(i,1:2), l2g(i,1:2)) + Ai(1:2,1:2);
    b(l2g(i,1:2)) = b(l2g(i,1:2)) + bi(1:2);
end
% condition aux limites
ud=zeros(Nso,1); ud(1)=g(x(1)); ud(Nso)=g(x(Nso));
b=b-A*ud;
b(1) = ud(1);b(Nso)=ud(Nso);
A(:,1) =0;A(1,:) =0;A(:,Nso) =0;A(Nso,:) =0;
A(1,1) =1;A(Nso,Nso) =1;
```

Q-3 : Constructions pour P2-Lagrange

1. Matrice et vecteur lmentaires

```
function [Ai,bi]=elementconvdiffP2(x1,x2,epsi,i)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [Ai,bi]=elementconvdiffP2(x1,x2,epsi,i)
% Assemble la matrice élémentaire et le second membre élémentaire
% du problème de convection diffusion pour P2-Lagrange
%
% Entrees:[x1,x2] intervalle
%         epsi coefficien epsilon
%         i numero de l'élément
% Sortie : Ai matrice 3 x 3
%         bi vecteur 3 x 1
%
%  COURS D'ELEMENTS FINIS
%  M2PRO 2007-2008    Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (July 2007)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%formule de quadrature de simpson
w = [1, 4, 1]*(x2-x1)/6; %poids
z=[x1 , (x1+x2)/2, x2]; %points
%initialisation
Ai=zeros(3,3);bi=zeros(3,1);
%on boucle sur les poind de quadrature
for q=1:3
    wq=w(q);
    zq = z(q);
    %boucle sur les lignes
    for l=1:3
        [vl,dvl] = baselocalP2(x1,x2,l,zq);
        bi(l)= bi(l) + wq * f(zq)*vl;
        %boucle sur les colonnes
        for m=1:3
            [vm,dvm] = baselocalP2(x1,x2,m,zq);
            Ai(l,m)=Ai(l,m) + wq * (epsi*dvl * dvm + vl*dvm);
        end
    end
end
```

2. Assemblage et introduction des conditions aux limites

```
function [A,b,y]=convdiffP2(epsi,x)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% function [A,b,y] =convdiffP2(x)
%
% fonction qui construit la matrice du probleme de convection diffusion
% pour les polynomes de degre 2 et introduit les conditions aux limites
%
% ENTREE: x les noeuds du maillage
%         eps le coefficient epsilon
%
% SORTIE: A matrice avec conditions aux limites incorporees
%         b vecteur second membre avec conditions aux limites incorporees
%         y position des degres de liberte
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Nso = length(x); % nombre de sommets
Nma = Nso -1 ;   % nombre de mailles
Ndl=Nso + Nma;   % nombre de degres de liberte
% table des degres de liberte
```

```
% et leur position
y=[];l2g=[];
for i = 1:Nma
    l2g(i,1) = i;
    l2g(i,2) = i+Nso;
    l2g(i,3) = i+1;
    y(l2g(i,1))=x(i);
    y(l2g(i,3)) = x(i+1);
    y(l2g(i,2)) = (x(i)+x(i+1))/2;
end
%reservation d'espace
A=zeros(Ndl,Ndl);
b=zeros(Ndl,1);
% boucle sur les elements
for i=1:Nma
    %On recupere les sommets de l'element
    x1 = x(i); x2 = x(i+1);
    % On calcule la matrice et le second membre elementaire
    [Ai,bi] = elementconvdiffP2(x1,x2,epsi,i);
    % et on assemble la matrice et le vecteur elementaires dans la
    % matrice globale et le vecteur global
    A(l2g(i,1:3), l2g(i,1:3)) = A(l2g(i,1:3), l2g(i,1:3)) + Ai(1:3,1:3);
    b(l2g(i,1:3)) = b(l2g(i,1:3)) + bi(1:3);
end
% prise en compte des conditions aux limites
ud=zeros(Ndl,1);ud(1)=g(x(1));ud(Nso)=g(x(Nso));
b=b-A*ud;
b(1) = ud(1);b(Nso)=ud(Nso);
A(:,1) =0;A(1,:) =0;A(:,Nso) =0;A(Nso,:) =0;
A(1,1) =1;A(Nso,Nso) =1;
```

Q-4 : Comparaisons

On commence par crire un script pour la solution exacte :

Solution exacte

```
function [y] = gexact(x,epsi)
y=x - (exp(x/epsi)-1)/(exp(1/epsi)-1);
```

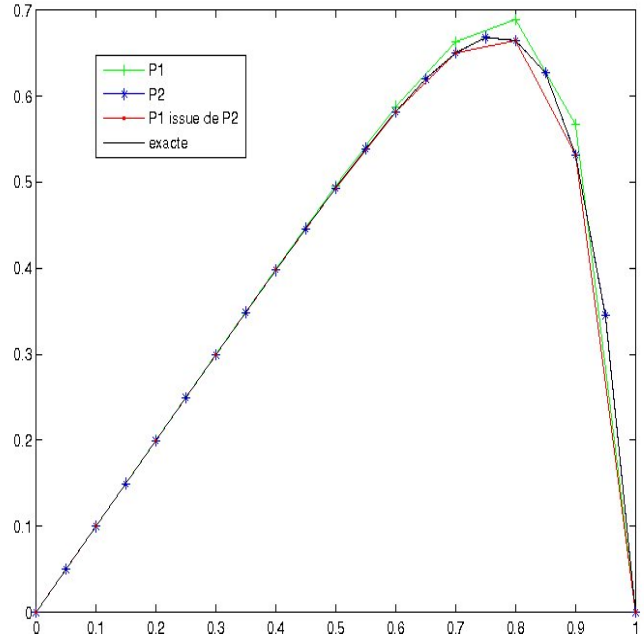
On introduit ensuite la fonction qui ralise le test pour un maillage x donn et une valeur de ϵ donne.

Script de comparaison

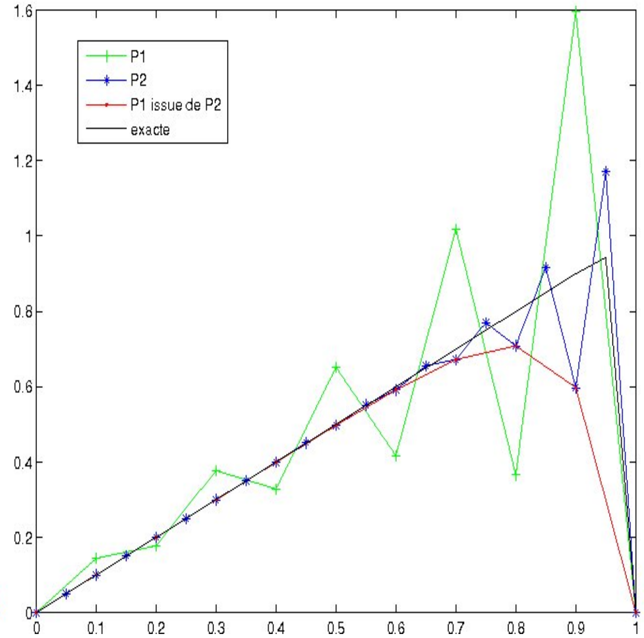
```
function  testconvdiff(x,epsi)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  testconvdiff(x,epsi)
%
% fonction qui compare les discretisation P1, P2 Lagrange
% de l'equation de convection diffusion
% ENTREE: x   -> les noeuds du maillage
%        epsi -> valeur de epsilon
% SORTIE:
% %  COURS MA0
%  M1 CALCUL SCIENTIFIQUE 2008-2009  Lab AN-EDP Univ d'ORSAY
% (c) Jean-Baptiste Apoung Kamga (Mars 2009)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%resolution P1
[A,b]=convdiffP1(epsi,x);
u=A¿
%resolution P2
[A2,b2,y]=convdiffP2(epsi,x);
u2=A2¿2;
[dum,p]=sort(y);
%solution exacte
ue=gexact(y,epsi);
%representation graphique
plot(x,u,'g+-',y(p),u2(p),'b*-',x,u2(1:length(x)),'r.-',y(p),ue(p),'k-');
legend('P1','P2','P1 issue de P2','exacte');
pause();
```

Reprsentation graphique de la solution exacte et la solution discrte sur les deux maillages.

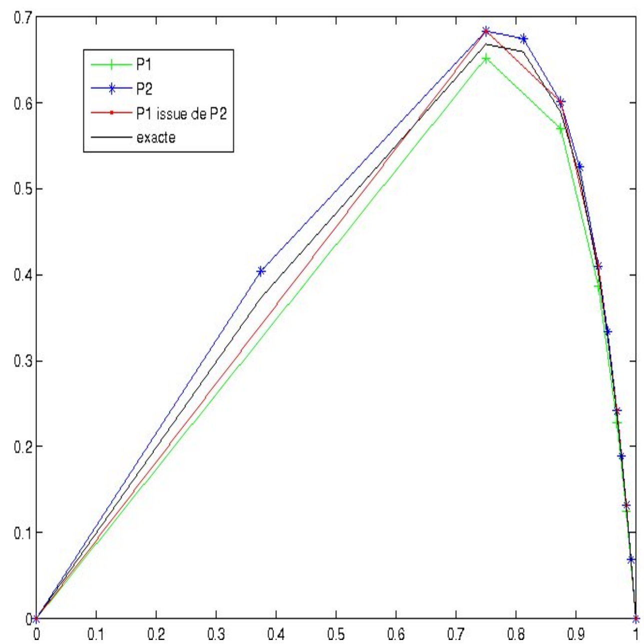


x=0:0.1:1; ϵ = 0.1.

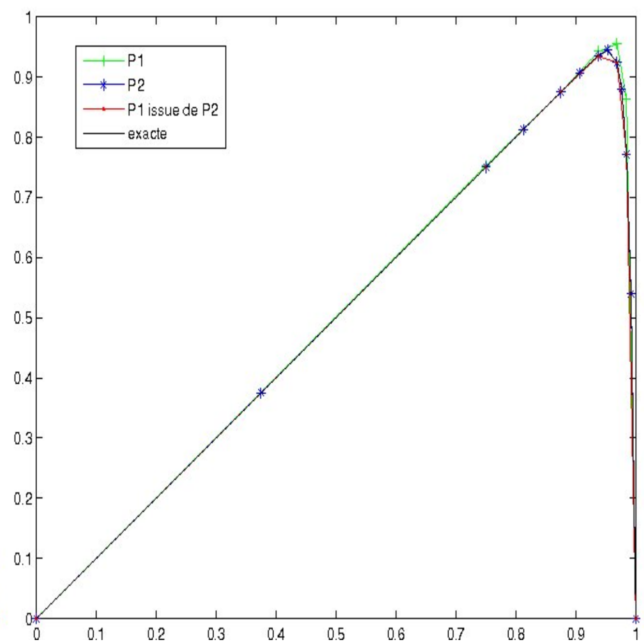


x=0:0.1:1; ϵ = 0.01.

On reprend avec `x = [0, 0.75,0.875, 0.9375, 0.96875, 0.984375, 1];`.



ϵ = 0.1



ϵ = 0.01

Vous pouvez explorer les points suivants.

1. **Sur le maillage uniforme dont h est du mme ordre que ϵ, il y a un bon comportement des lments finis Lagrange. [*renseignez-vous sur le nombre de Pclet*].**
2. **L'interpol P1 de la solution P2 prsente moins d'oscillations et semble mieux se comporter mme lorsque la solution P2 oscille.**
3. **Le raffinement de maillage au voisinage de $x = 1$ conduit un bon comportement des lments finis Lagrange.[*renseignez-vous sur la couche limite*].**