Quelques aspects pratiques

de la

méthode des Eléments Finis

Maillage

Création de maillages triangulaires : on peut utiliser le logiciel `emc2`

Structure d'un fichier `toto.amdba` généré par `emc2`

```
 21 28  -- nbs,nbt
         1   1.8462181E-02  -3.8533497E+00        0
         2  -3.8346165E-01  -2.3533497E+00        1
         3  -1.4815379E+00  -1.2552733E+00        1
        ....
        20  -1.8798810E+00  -3.8135126E+00        0
        21  -2.4839835E+00  -2.4373295E+00        0
     1     10     14      9        1
     2     15     20     14        1
     3     18     13     17        1
    ....
    27     21      3      4        1
    28     18     21     19        1
```

Maillage

Structure d'un fichier `toto.amdba` généré par `emc2`

```
Nbre_noeuds  Nbre_triangles
Pour i = 1, ..., Nbre_noeuds
  i  Coor_Noeud(i,1)  Coor_Noeud(i,2)  Ref_noeud(i)
Pour l = 1, ..., Nbre_triangles
  l   Num_Sommet(l,1)  Num_Sommet(l,2)  Num_Sommet(l,3)  Ref_triangle(l)
```

`Ref_noeud(i)`	indique si le noeud `i` est un noeud de frontière ou pas
`Ref_triangle(l)`	indique dans quel domaine se trouve le triangle `l`

Maillage

Structure d'un fichier `toto.msh` généré par `emc2`

```
21 28 12
  1.846218E-02  -3.853350E+00       0
 -3.834617E-01  -2.353350E+00       1
 -1.481538E+00  -1.255273E+00       1
 ...
 -1.879881E+00  -3.813513E+00       0
 -2.483984E+00  -2.437330E+00       0
    10      14       9        1
    15      20      14        1
    18      13      17        1
    ...
    21       3       4        1
    18      21      19        1
     1       2       1
     2       3       1
    ....
    11      12       1
    12       1       1
```

Maillage

Structure d'un fichier toto.msh généré par emc2

```
Nbre_noeuds  Nbre_triangles  Nbre_aretes_interfaces
Pour i = 1, ..., Nbre_noeuds
     Coor_Noeud(i,1)  Coor_Noeud(i,2)  Ref_noeud(i)
Pour l = 1, ..., Nbre_triangles
     Num_Sommet(l,1)  Num_Sommet(l,2)  Num_Sommet(l,3)  Ref_triangle(l)
Pour k = 1, ..., Nbre_aretes_interfaces
     Num_Sommet_A(k,1)  Num_Sommet_A(k,2)  Ref_arete(k)
```

Structure d'un programme Eléments Finis

Etape 1 Lecture du maillage.

a - Lecture des tableaux :

```
Coor_Noeud(1:Nbre_noeuds, 1:2)
Num_Sommet(1:Nbre_triangles, 1:3)
Ref_noeud(1:Nbre_noeuds)
...
```

b - Création des degrés de libertés (ddls)

```
Coor_ddl(1:Nbre_ddls, 1:2)
Num_ddl(1:Nbre_triangles, 1:N_ddls_triangle)
```

(*numérotation globale des ddls par triangles*)

Remarque : Pour les éléments finis P_1 :

```
Coor_ddl = Coor_Noeud  ;  Num_ddl = Num_Sommet
```

Structure d'un programme Eléments Finis

Etape 2 : Calcul et assemblage de la matrice : $A_{I,J} = a(w_J, w_I)$

`A = sparse(Nbre_ddls, Nbre_ddls)` *(déf. matrices creuses sous matlab)*

```
Pour l = 1, Nbre_triangles (boucle sur les triangles)
     Pour i = 1, Nbre_ddls_triangle
          Pour j = 1, Nbre_ddls_triangle
               Calcul des A^l(i,j) = a(τ_j^l, τ_i^l)
               I = Num_ddl(l, i)
               J = Num_ddl(l, j)
               A_{I,J} = A_{I,J} + A^l(i,j)
          Fin boucle j
     Fin boucle i
 Fin boucle l
```

Structure d'un programme Eléments Finis

Etape 3 : Calcul et assemblage du second membre : $F_I = \ell(w_I)$

F = 0

Pour l = 1, Nbre_triangles *(boucle sur les triangles)*

Pour i = 1, Nbre_ddls_triangle

Calcul de $F_i^l = \ell(\tau_i^l)$

I = Num_ddl(l, i)

$F_I = F_I + F_i^l$

Fin boucle i

Fin boucle l

Structure d'un programme Eléments Finis

Etape 4 : Prise en compte des conditions aux limites de Dirichlet

Pour les ddls I qui sont sur un bord de Dirichlet

- on met à zéro le second membre F_I

- on met à zéro les termes extra-diagonaux sur la I ème ligne et la I ème colonne de A.

```
Pour I = 1, Nbre_ddls
     IF (ref_ddl(I) == ref bord de Dirichlet) THEN
```

$$F_I = 0$$

$$A_{I,J} = A_{J,I} = \delta_{I,J} A_{I,I}$$

```
     END
Fin boucle I
```

Structure d'un programme Eléments Finis

Etape 5 : Résolution du système linéaire :

$$AU = F$$

On résout le système sans calculer l'inverse de la matrice !

Exemple : par élimination de Gauss (factorisation LU)

`U = A \ F ;` *(sous matlab)*

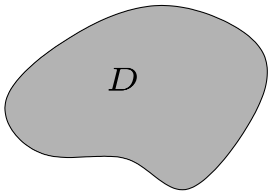
Etape 6 : Ecriture (et visualisation) de la solution

Exemple sous matlab

```
trisurf(Num_Sommet(:,1:3),Coor_Noeud(:,1),Coor_Noeud(:,2), U);
view(2);
shading interp;
```

Résolution d'un problème de diffraction

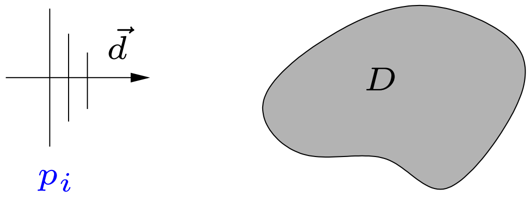
$$\begin{cases} \Delta p + k^2 p = 0 & \text{dans } \mathbb{R}^2 \setminus D \\ p = 0 & \text{sur } \partial D \end{cases}$$



On décompose la solution en la somme : $p = p_i + p_s$

Résolution d'un problème de diffraction

$$\begin{cases} \Delta p + k^2 p = 0 & \text{dans } \mathbb{R}^2 \setminus D \\ p = 0 & \text{sur } \partial D \end{cases}$$



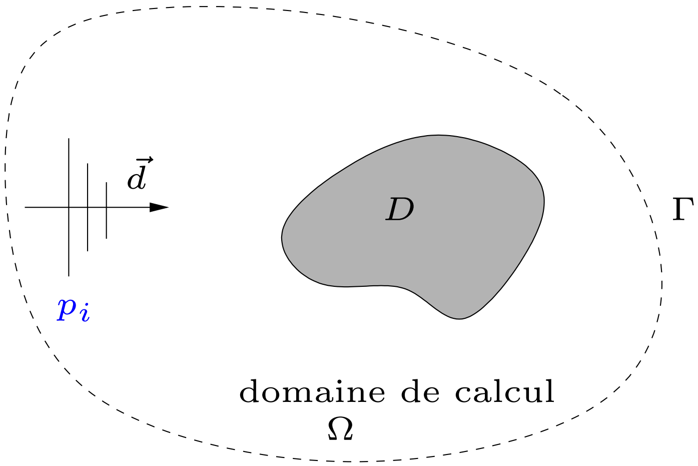
On décompose la solution en la somme : $p = p_i + p_s$

p_i : onde incidente (par exemple, une onde plane : $p_i(\vec{x}) = e^{ik\vec{x}\cdot\vec{d}}$; $|\vec{d}| = 1$).

p_s : onde diffractée

Résolution d'un problème de diffraction

$$\begin{cases} \Delta p + k^2 p = 0 & \text{dans } \mathbb{R}^2 \setminus D \\ p = 0 & \text{sur } \partial D \end{cases}$$



On décompose la solution en la somme : $p = p_i + p_s$

p_i : onde incidente (par exemple, une onde plane : $p_i(\vec{x}) = e^{ik\vec{x}\cdot\vec{d}}$; $|\vec{d}| = 1$).

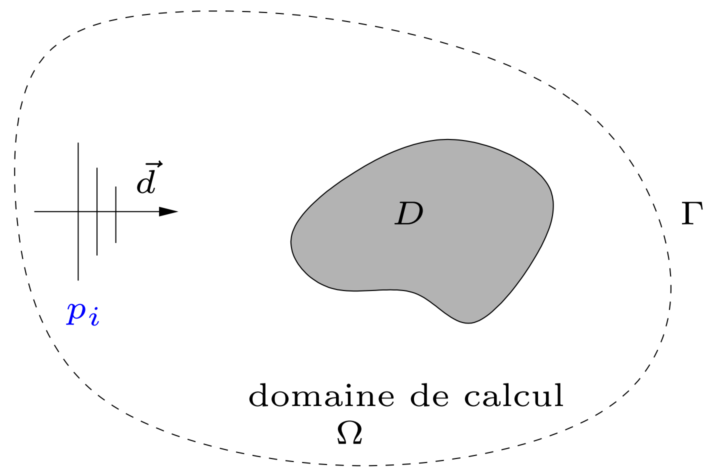
p_s : onde diffractée

On borne le domaine de calcul par une frontière fictive Γ

Condition d'onde sortante : $\dfrac{\partial p_s}{\partial n} - ikp_s = 0 \quad$ sur Γ

Résolution d'un problème de diffraction

$$\begin{cases} \Delta p_s + k^2 p_s = 0 & \text{dans } \Omega \\ p_s = -p_i & \text{sur } \partial D \\ \frac{\partial p_s}{\partial n} - ikp_s = 0 & \text{sur } \Gamma \end{cases}$$



On décompose la solution en la somme : $p = p_i + p_s$

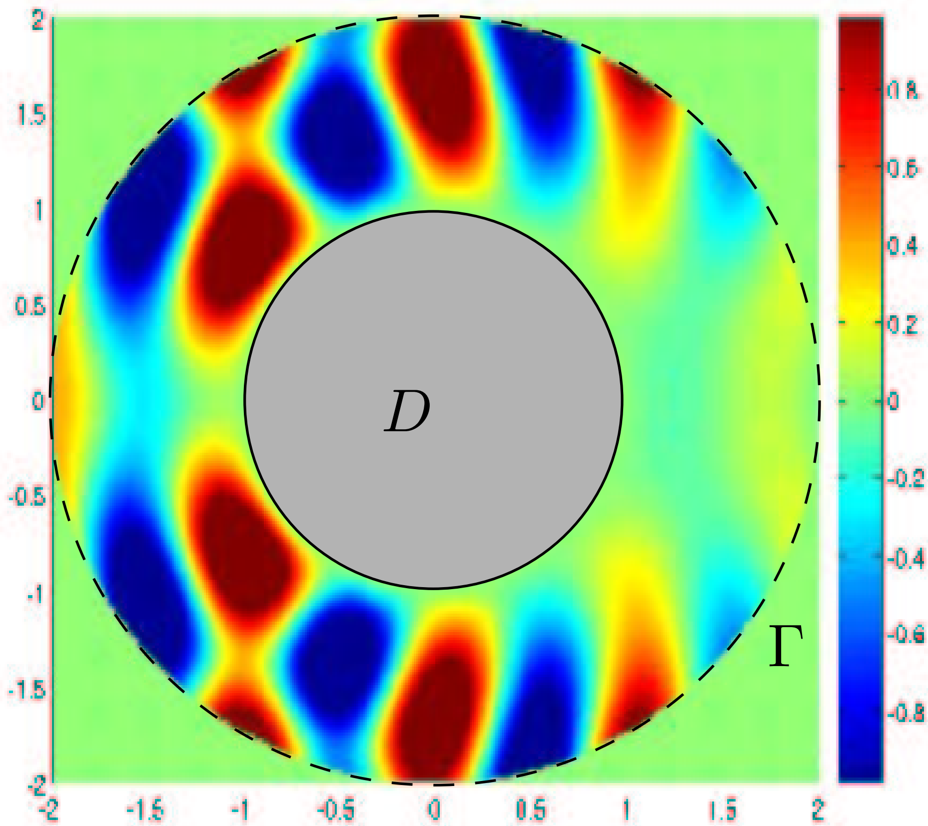
p_i : onde incidente (par exemple, une onde plane : $p_i(\vec{x}) = e^{ik\vec{x}\cdot\vec{d}}$; $|\vec{d}| = 1$).

p_s : onde diffractée

On borne le domaine de calcul par une frontière fictive Γ

Condition d'onde sortante : $\dfrac{\partial p_s}{\partial n} - ikp_s = 0 \quad$ sur Γ

Exemple de géométries régulières



Partie réelle de la solution (E.F. P2)

Maillage

Courbes de convergence en géométries régulières

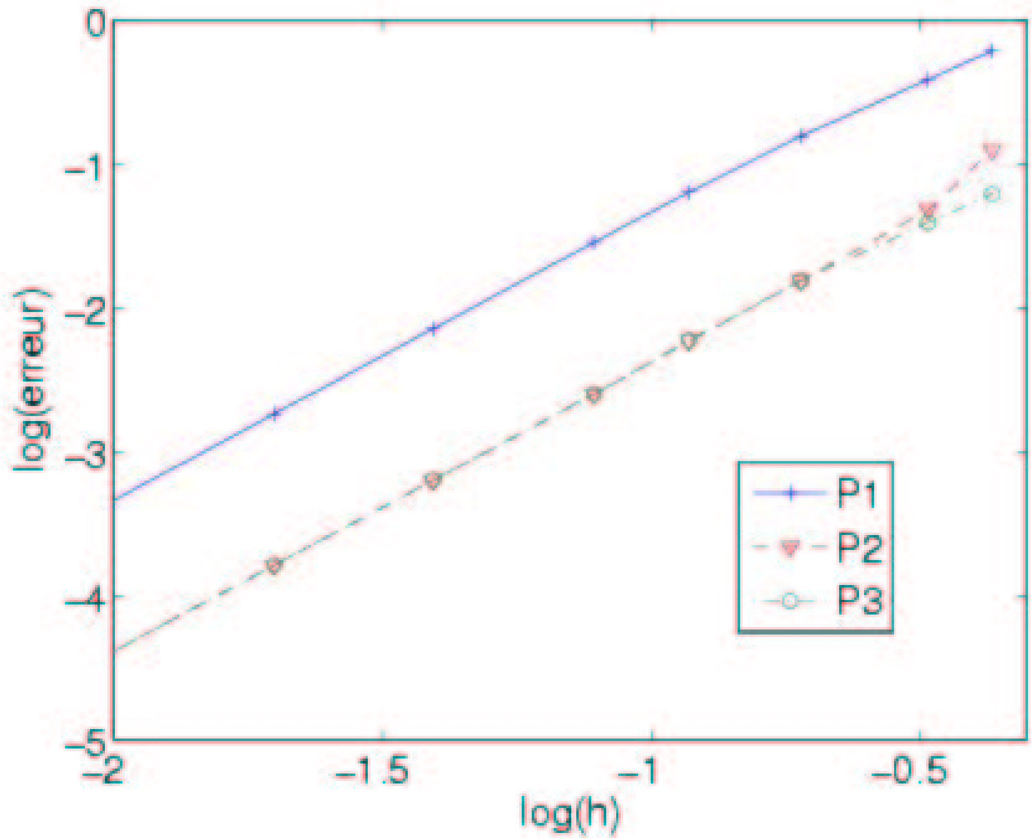
Rappel : si p_h^k est la solution approchée calculée avec les E.F. P_k,

$$\|p - p_h^k\|_{H^1(\Omega)} \underset{h \to 0}{\sim} C\, h^k\, \|p\|_{H^{k+1}(\Omega)}$$

$$\log\left(\|p - p_h^k\|_{H^1(\Omega)}\right) \underset{h \to 0}{\sim} k\, \log(h) + \log(C\|p\|_{H^{k+1}(\Omega)})$$

$$\log(h) \mapsto \log\left(\|p - p_h^k\|_{H^1(\Omega)}\right) \quad \underset{h \to 0}{\sim} \quad \text{droite de pente } k$$

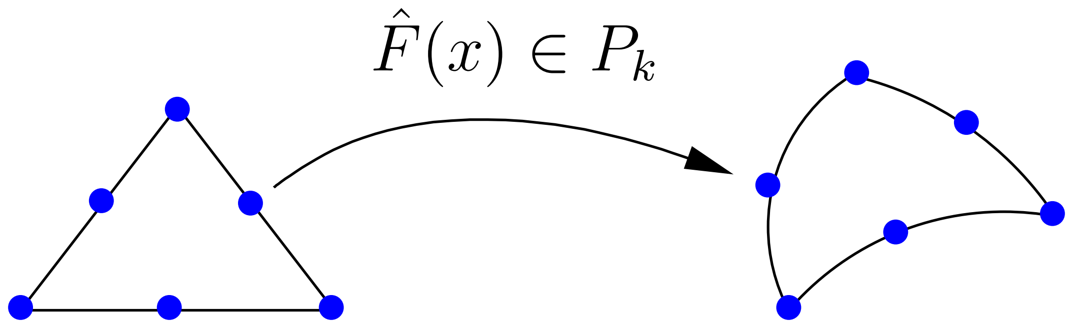
Courbes de convergence pour des géométries régulières



E.F. de Lagrange P_k

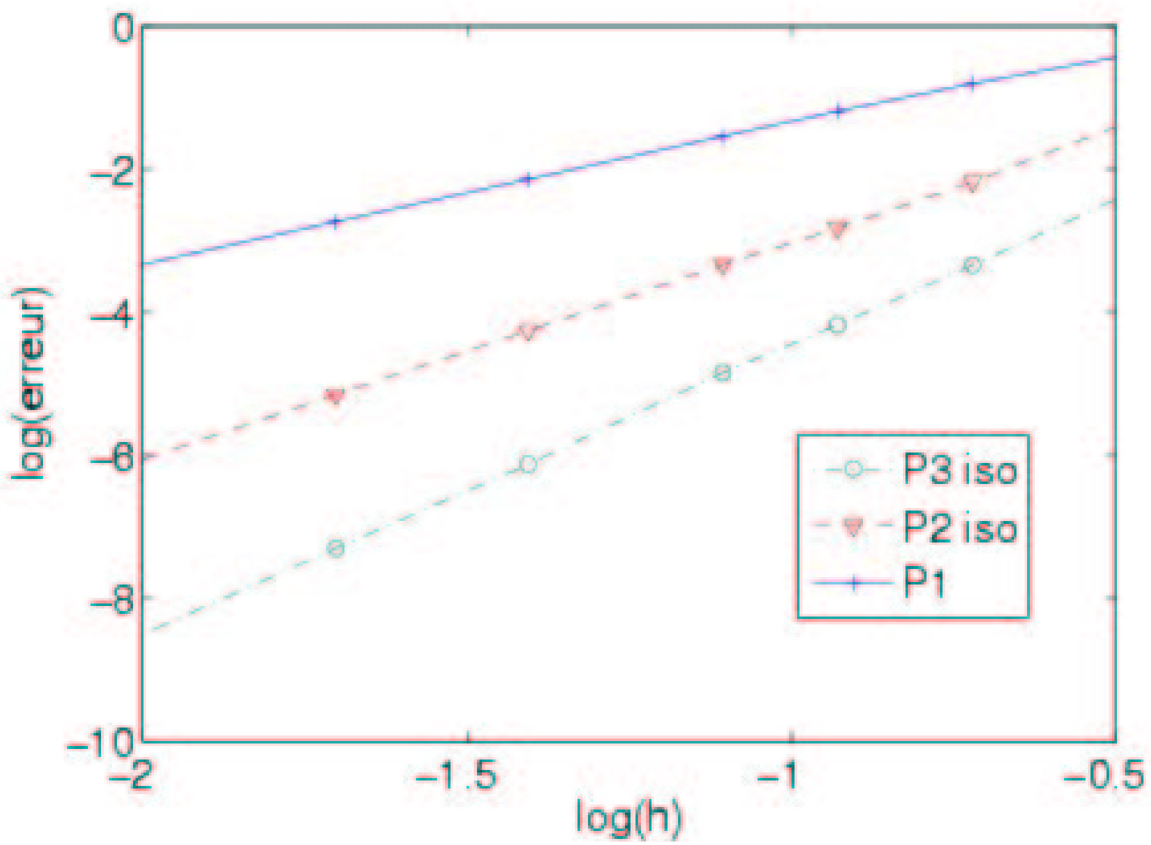
L'erreur d'approximation géométrique empèche l'obtention d'une convergence en $O(h^k)$

E.F. isoparamétriques



E.F. de référence E.F. isoparamétrique

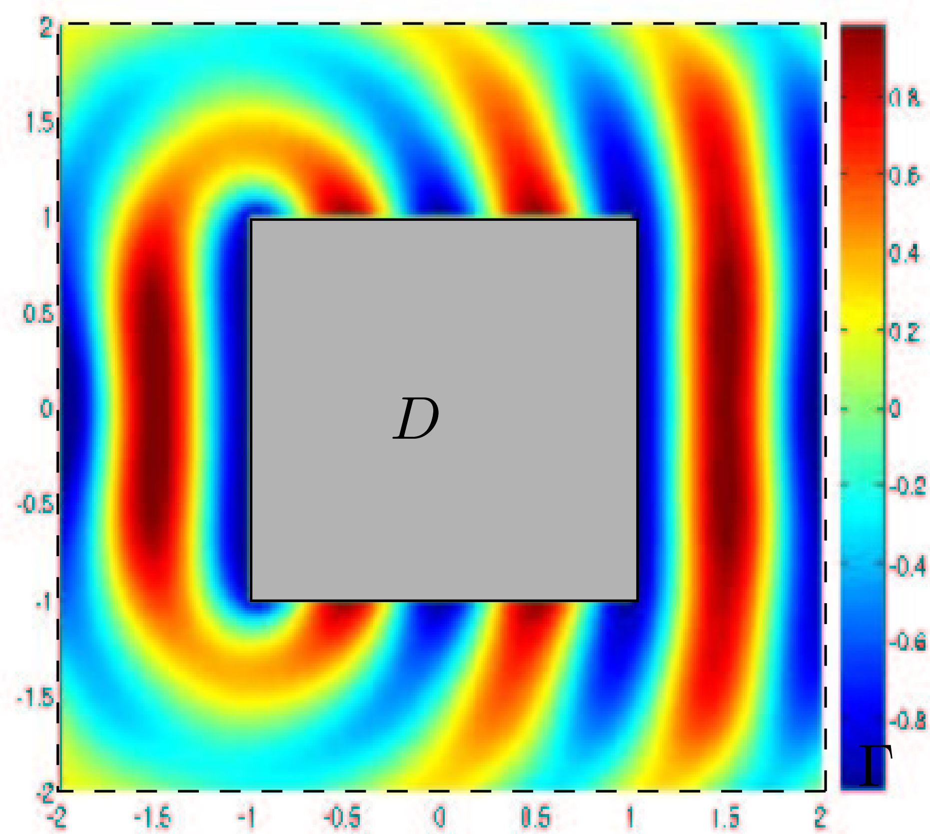
Courbes de convergence pour les E.F. iso.



E.F. courbes (isoparamétriques)

On récupère un odre de convergence en $O(h^k)$

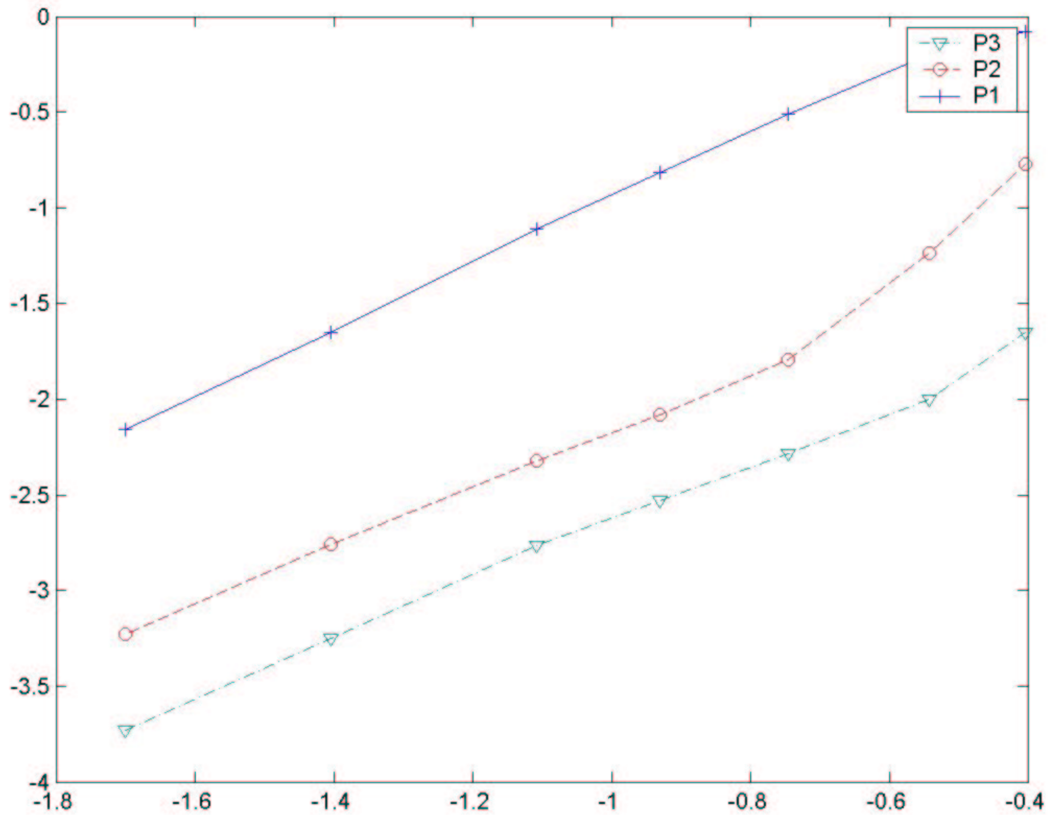
Exemple de géométries non régulières



Module du champ diffracté (E.F. P2)

Maillage

Courbes de convergence en présence de coins



E.F. courbes (isoparamétriques)

La solution $\notin H^s$, $s \geq 2 \Rightarrow$ même vitesse de convergence pour les E.F, P_k, $k > 1$.