CEA–EDF–Inria Summer School June 27 – July 1, 2022
Certification of errors in numerical simulations EDF Lab, Paris-Saclay
https://ecoles-cea-edf-inria.fr/en/schools/ecole-analyse-numerique-2022/
Freefem++ web page: https://freefem.org/

Computer tutorial N°4

Convergence and optimality of the adaptive FEM
Mesh adaptation, rate of convergence with respect to the number of degrees of fredom, optimal (best-possible) decay

Let $\Omega \subset \mathbb{R}^2$ be a polygon with Lipschitz boundary $\partial\Omega = \Gamma_{\rm D} \cup \Gamma_{\rm N}$. We consider the following model problem: for a given source term $f \in L^2(\Omega)$ and a given prescribed data $g_{\rm D}$ on the Dirichlet part of the boundary $\Gamma_{\rm D}$, find $u : \Omega \to \mathbb{R}$ such that

$$-\Delta u = f \quad \text{in } \Omega, \qquad u = g_{\rm D} \text{ on } \Gamma_{\rm D}. \tag{1}$$

Exercices 1 below is designed for the case where $\Omega = (0,1)^2$, $\Gamma_{\rm D} = \partial\Omega$, $g_{\rm D} = 0$, and $f = -2(x^2+y^2) + 2(x+y)$. In this case,

$$u(x,y) = x(x-1)y(y-1), \tag{2}$$

which is a smooth solution.
Exercice 2 and Exercice 3 present extension to the L-shaped domain $\Omega = (-1,1) \times (-1,1) \setminus [0,1] \times [-1,0]$ together with the exact solution written, in polar coordinates with $\theta \in (0, 3\pi/2)$, as

$$u(r,\theta) = r^{\frac{2}{3}} \sin(2\theta/3). \tag{3}$$

We remark that the exact solution is singular here, $u \in H^{\frac{5}{3}-\varepsilon}(\Omega)$ for arbitrarily small $\varepsilon > 0$. The corresponding source term $f = 0$, and we take $g_{\rm D} = u$ on $\Gamma_{\rm D} = \partial\Omega$.

Exercice 1. (Error and estimator on uniformly refined meshes for smooth example)

1. Compute the "P1" FE approximation and the "RT1" equilibrated flux. Set up the following parameters:

```
int nds = 4; // number of mesh points on one domain unity edge
FinalLevel = 4; // Number of level of mesh refinement
bool RunAdaptive = 0; // 0 means uniform mesh, 1 means adaptive mesh
bool RunSmooth = 1; // 0 means singular example, 1 means smooth example
```

   Remove the comment /* and define the functions of the PDEs data for the smooth example:

```
///*
// CASE 1 (smooth polynomial in a unit square)
func uEx = x*(x-1)*y*(y-1);
func dxuEx = (2*x-1)*y*(y-1);
......
real aevx=0.5, aevy=0.5;
int trdet1=88, trdet2=91;
//*/
```

2. Plot the convergence line for the error and estimator against the total number of DoFs.

3. Check the command window what is the convergence rate and effectivity index on the uniformly refined meshes sequence. Is the convergence rate optimal?

4. Compute the "P2" FE approximation and the "RT2" equilibrated flux. Check the convergence rate on the uniform refined meshes sequence.

Answer 1. (Error and estimator on uniformly refined meshes for singular example)

We take `nds=4`, `FinalLevel = 4`, `bool RunAdaptive = 0` and `bool RunSmooth = 1`.

1. One should obtain the results as in Figures 1 and 2. In addition, one should observe that the FEM solution is more and more accurate on uniformly refined meshes.

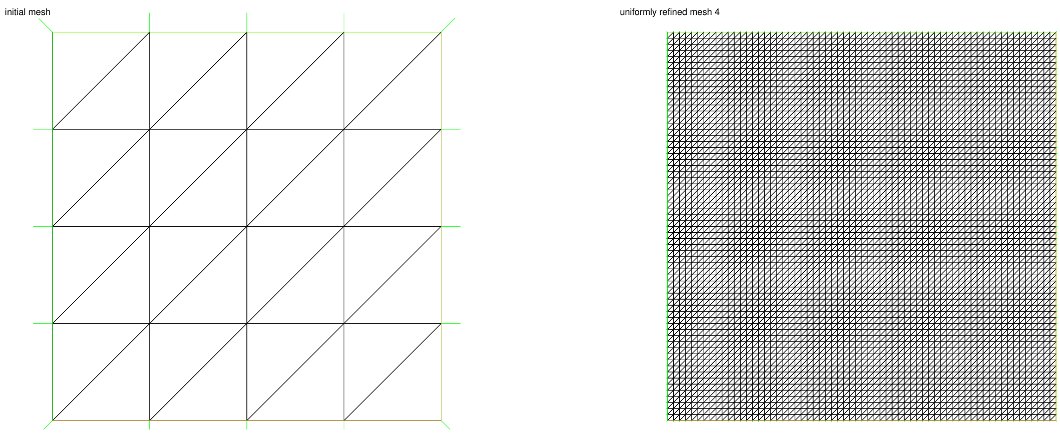


Figure 1: Initial mesh (left) and finest refined mesh (right)

2. One should first get into the file containing the data generated by the code `TP4.edp`, run the following command:

   `gnuplot ConvergenceRate.plt; // plot the datum`

   Then, you should see the following convergence Figure 3 in the loglog scale. The x-axis is the total number of Degrees of Freedoms (DoFs) and y-axis is the energy norm error and error estimator. We know that in this case of a smooth solution, the convergence rate is $\mathcal{O}(h^p)$, which becomes $\mathcal{O}(\text{DoFs}^{-\frac{p}{2}})$ in terms of DoFs. So, in the present case with $p = 1$, we need to find $\mathcal{O}(\text{DoFs}^{-\frac{1}{2}})$.

3. Going back to the command window, one should find the convergence rate and effectivity index the numerical solution computed on the sequence of uniformly refined meshes. You should see the following data:

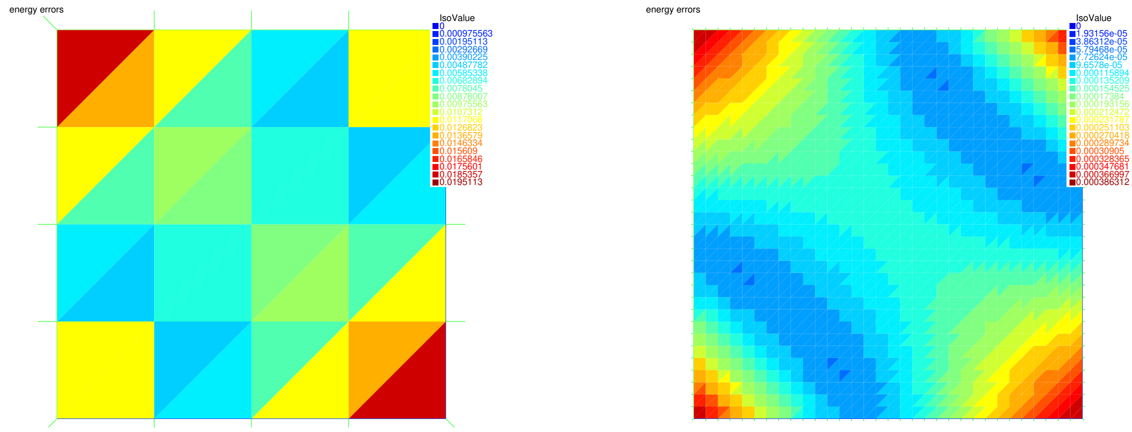
   `Convergence rate 3`

   `0.5675315829 0.5397301454 0.5212486906`

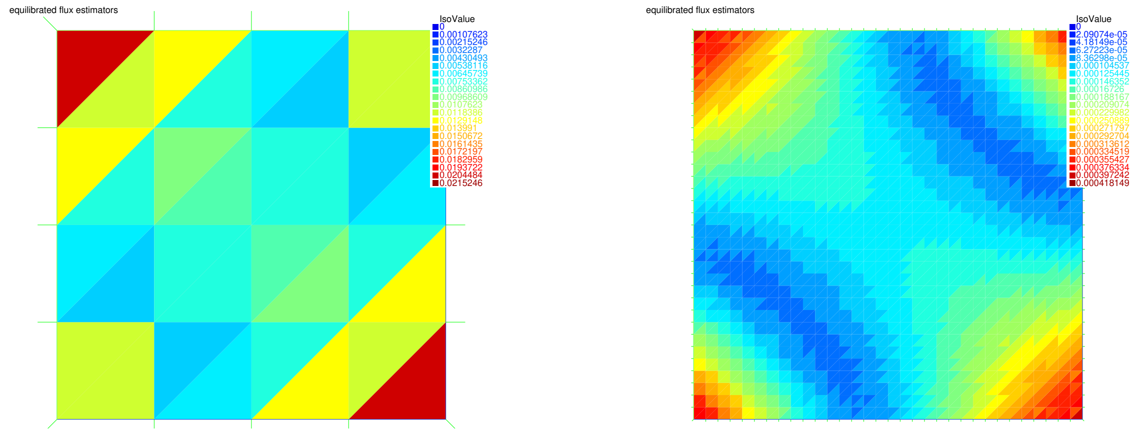
   `Effectivity Index 4`

   `1.044909373 1.043766139 1.045481868 1.046058539`

   The above data shows the convergence rate for FEM with "P1" basis under the uniform mesh refinement is $\mathcal{O}(\text{DoFs}^{-\frac{1}{2}})$, which is optimal in terms of DoFs. In addition, the effectivity index converges to 1.04.



(a) Elementwise errors $\|\nabla(u - u_\ell)\|_K$



(b) Elementwise estimators $\|\nabla u_\ell + \boldsymbol{\sigma}_\ell\|_K$

Figure 2: Initial mesh (left) and finest-refined mesh (right)

4. One should set up “P2” FE approximation and the “RT2” equilibrated flux, and then run the code again. You should see the following data in the command window:

   ```
   Convergence rate 3
   1.073175248 1.040710629 1.021428279
   Effectivity Index 4
   1.019698545 1.013056183 1.009550153 1.007757069
   ```

   The above data shows the convergence rate for FEM with “P2” basis under the uniform meshes refinement is $\mathcal{O}(\text{DoFs}^{-1})$, which is optimal in terms of DoFs. In addition, the effectivity index converge to 1.007. One can also plot the convergence Figure 4 for the FEM with “P2” basis.

   So the above numerical results of the FEM confirm the theoretical results that the convergence rate of FEM for the smooth solution is optimal in terms of DoFs with the rate $\mathcal{O}(\text{DoFs}^{-\frac{p}{2}})$ (i.e., $\mathcal{O}(h^p)$ in the more common terms) with any order $p \geq 1$ under the uniform mesh refinement.

Exercice 2. (Error and estimator on uniformly refined meshes for singular example)

1. Compute the “P1” FE approximation and the “RT1” equilibrated flux. Set up the following parameters:

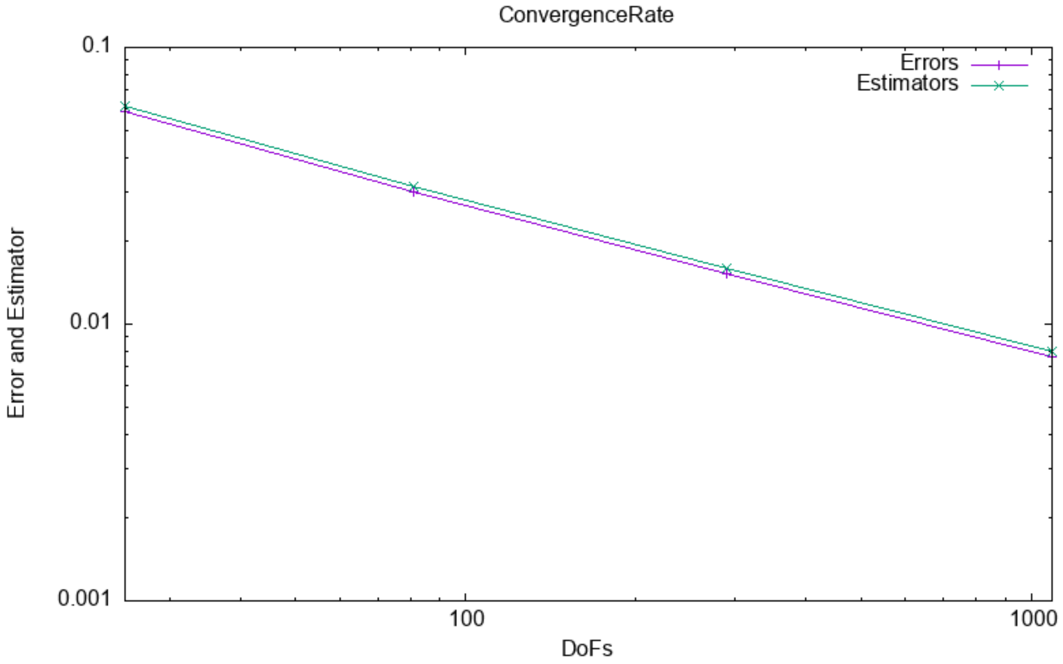


Figure 3: Convergence of the energy error and estimator under uniformly refined meshes with P1 basis

```
int nds = 4; // number of mesh points on one domain unity edge
FinalLevel = 4; // Number of level of mesh refinement
bool RunAdaptive = 0; // 0 means uniform mesh, 1 means adaptive mesh
bool RunSmooth = 0; // 0 means singular example, 1 means smooth example
```

Remove the comment `/*` and define the functions of the PDEs data for the singular example:

```
///*
// CASE 2 (vertex singularity)
func theta=atan2(y, x)-2*pi*fmin(sign(y),0);
func r=(x^2+y^2)^(1/2.0);
......
mesh Th = buildmesh(b1(nds) + b2(nds) + b3(2*nds) + b4(2*nds)
+ b5(nds) + b6(nds));
//*/
```

2. Plot the convergence line for the error and estimator against the total number of DoFs.

3. Check the command window what is the convergence rate and effectivity index on the uniformly refined meshes sequence. Is the convergence rate optimal?

4. Compute the "P2" FE approximation and the "RT2" equilibrated flux. Check the convergence rate on the uniformly refined meshes sequence.

Answer 2. (Error and estimator on uniformly refined meshes for singular example)

We take `nds=4`, `FinalLevel = 4`, `bool RunAdaptive = 0` and `bool RunSmooth = 0`.

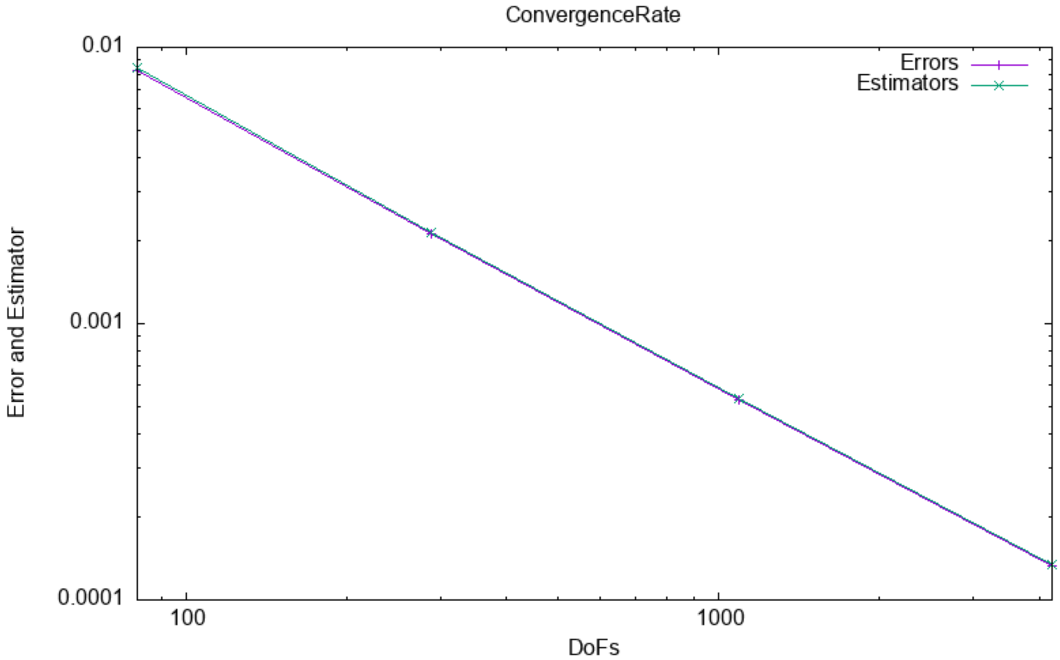


Figure 4: Convergence of the energy error and estimator under uniformly refined meshes with P2 basis

1. One should obtain the results as in Figures 5 and 6. In addition, one should observe that the FEM solution is more and more accurate on uniformly refined meshes. The error and estimator both take large values close to the origin.

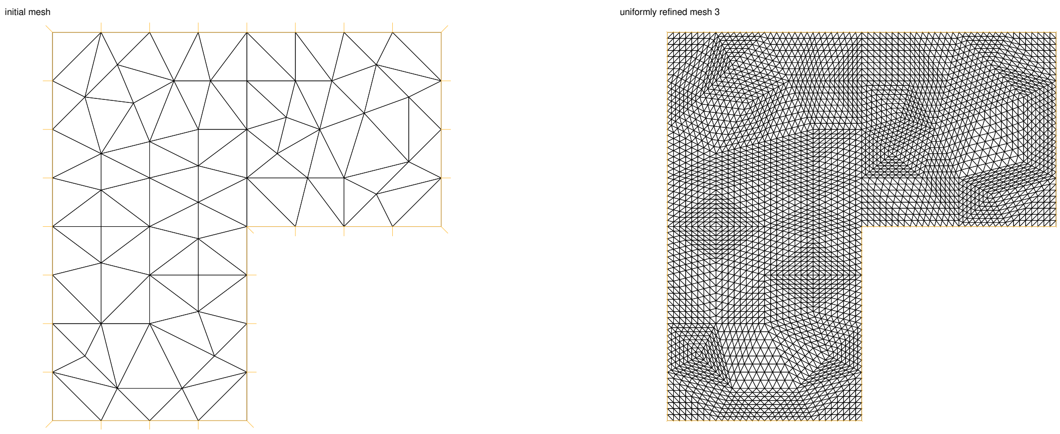


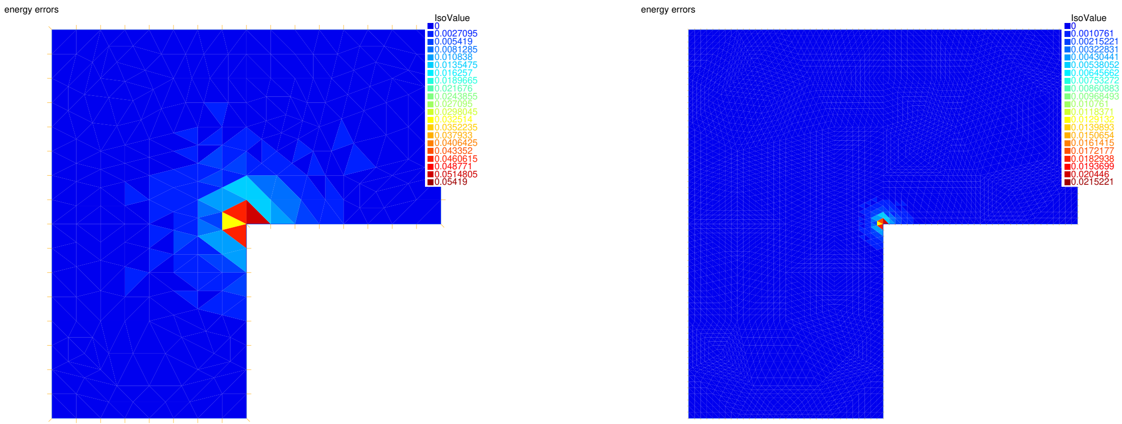
Figure 5: Initial mesh (left) and finest refined mesh (right)

2. One should first get into the file containing the data generated by the code `TP4.edp`, run the following command:

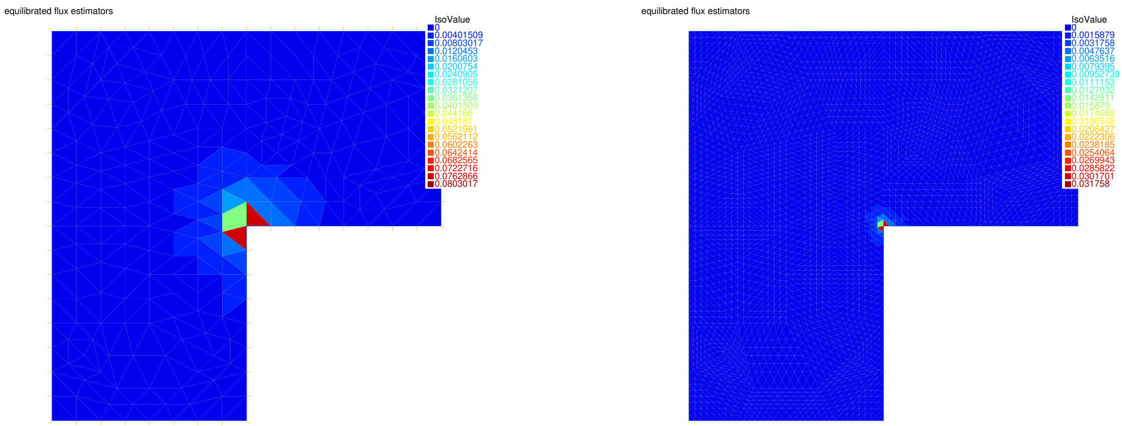
   ```
   gnuplot ConvergenceRate.plt; // plot the datum
   ```

   Then, you should see the following convergence Figure 7 in the log-log scale. The x-axis is the total number of Degrees of Freedoms (DoFs) and the y-axis is the energy norm error and error estimator.

3. Going back to the command window, one should find the convergence rate and effectivity index of the numerical solution computed on the sequence of uniformly refined meshes. You should see the following data:



(a) Elementwise errors $\|\nabla(u - u_\ell)\|_K$



(b) Elementwise estimator $\|\nabla u_\ell + \boldsymbol{\sigma}_\ell\|_K$

Figure 6: Initial mesh (left) and finest-refined mesh (right)

```
Convergence rate 3
0.3502560437 0.3397704118 0.3352352101
Effectivity Index 4
1.254142287 1.241282191 1.234310079 1.230385969
```

The above data shows the convergence rate for FEM with "P1" basis under the uniform meshes refinement is $\mathcal{O}(\mathrm{DoFs}^{-\frac{1}{3}})$ (this is $\mathcal{O}(h^{2/3})$ in terms of the mesh size h since, recall, $u \in H^{\frac{5}{3}-\varepsilon}(\Omega)$ only). This rate is not optimal in terms of DoFs. In addition, the effectivity index converges to 1.23.

4. One should set up the "P2" FE approximation and the "RT2" equilibrated flux and then run the code again. You should see the following data in the command window:

```
Convergence rate 3
0.3494536439 0.3420287166 0.3377536761
Effectivity Index 4
1.387788749 1.385561257 1.385172685 1.385069477
```

The above data shows the convergence rate for FEM with "P2" basis under the uniform meshes refinement is still $\mathcal{O}(\mathrm{DoFs}^{-\frac{1}{3}})$ (i.e., $\mathcal{O}(h^{2/3})$ in terms of h), which is the same value as for $p = 1$ and again not optimal in terms of DoFs. In addition,

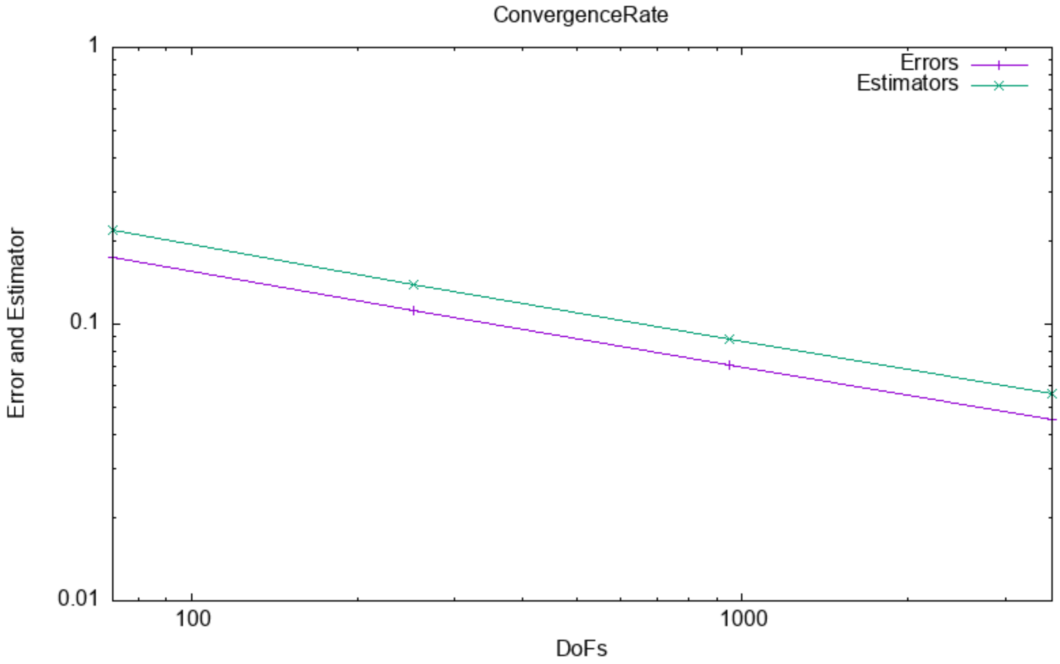


Figure 7: Convergence of the energy error and estimator under uniformly refined meshes with P1 basis

the effectivity index converge to 1.38. One can also plot the convergence Figure 8 for the FEM with “P2” basis.

So the above numerical results of the FEM confirm the theoretical results that the convergence rate of FEM for a singular solution is suboptimal for any order $p \geq 1$ of the basis in terms of DoFs under uniform mesh refinement: for uniform mesh refinement, we will always obtain $\mathcal{O}(\text{DoFs}^{-\frac{1}{3}})$ or $\mathcal{O}(h^{2/3})$, independently of the polynomial degree p. We will see that this will crucially change with proper mesh adaptation.

Exercice 3. (Error and estimator on adapted meshes for singular example)

1. Compute the “P1” FE approximation and the “RT1” equilibrated flux. Set up the following parameters:

```
int nds = 4; // number of mesh points on one domain unity edge
FinalLevel = 10; // Number of level of mesh refinement
bool RunAdaptive = 1; //  0 means uniform mesh, 1 means adaptive mesh
bool RunSmooth = 0; // 0 means singular example, 1 means smooth example
```

2. Plot the convergence line for the error and estimator against the total number of DoFs.

3. Check the command window what is the convergence rate and effectivity index on the adaptively generated meshes sequence. Is the convergence rate optimal?

4. Compute the “P2” FE approximation and the “RT2” equilibrated flux. Check the convergence rate on the uniform refined meshes sequence.

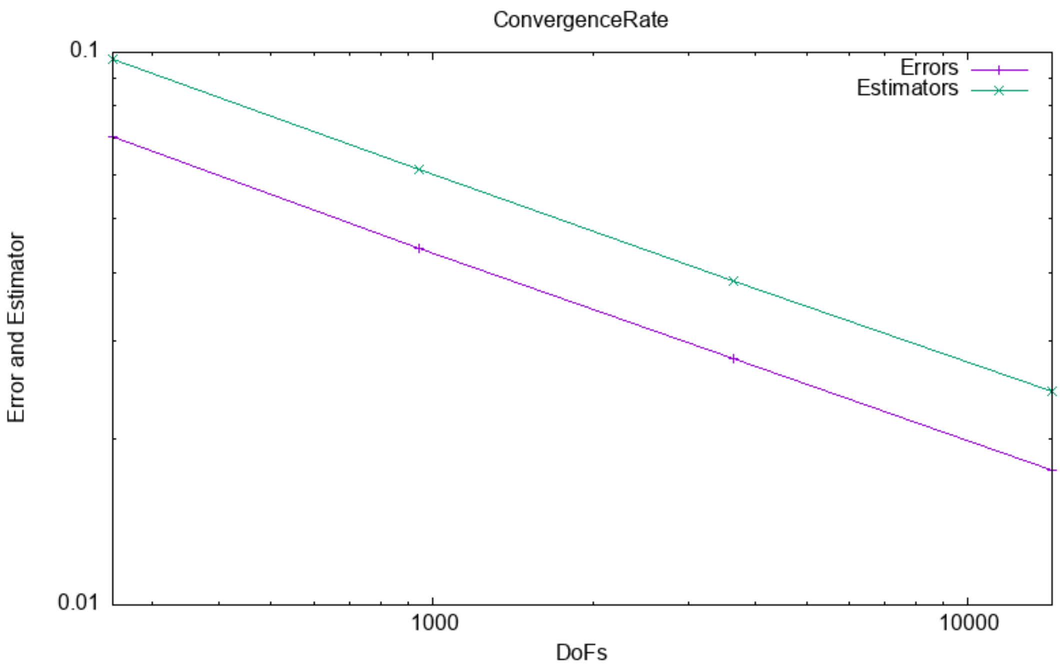


Figure 8: Convergence of the energy error and estimator under uniformly refined meshes with P2 basis

Answer 3. (Error and estimator on adaptive meshes for singular example)
We take `nds=4`, `FinalLevel = 10`, `bool RunAdaptive = 1` and `bool RunSmooth = 0`.

1. One should obtain a sequence of adaptively generated meshes as in Figure 9.

2. One should first get into the file containing the data generated by the code `TP4.edp`, run the following command:

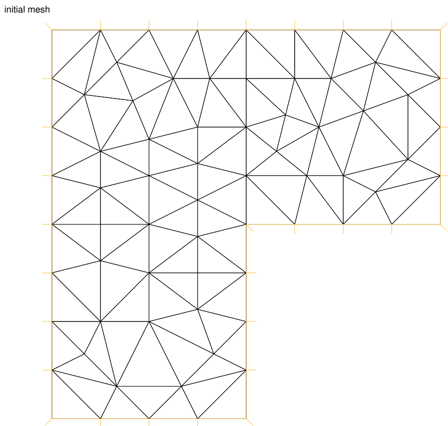
   ```
   gnuplot ConvergenceRate.plt; // plot the datum
   ```

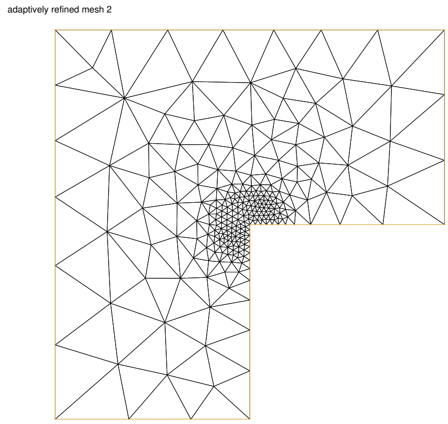
   Then, you should see the following convergence Figure 10 in the log-log scale. The x-axis is the total number of Degrees of Freedoms (DoFs) and the y-axis is the energy norm error and error estimator.

3. Going back to the command window, one should find the convergence rate and effectivity index of the numerical solution computed on the sequence of adaptively generated meshes. You should see the following data:

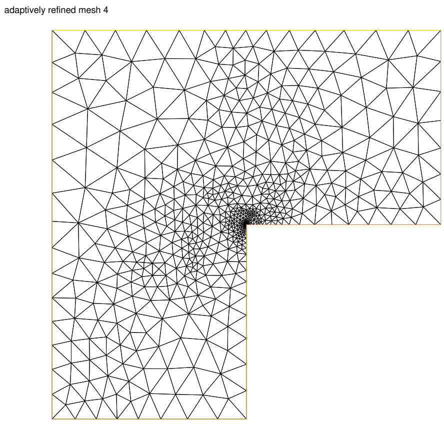
   ```
   Convergence rate 9

   1.090638799 0.4803245272 1.385556097 0.6239685494 0.4872797598

   0.5558842068 0.5004908804 0.5136038803 0.5333990767

   Effectivity Index 10

   1.254142287 1.18386325 1.109395379 1.073301097 1.070741775

   1.063848946 1.061339548 1.060667381 1.059648214 1.059516139
   ```

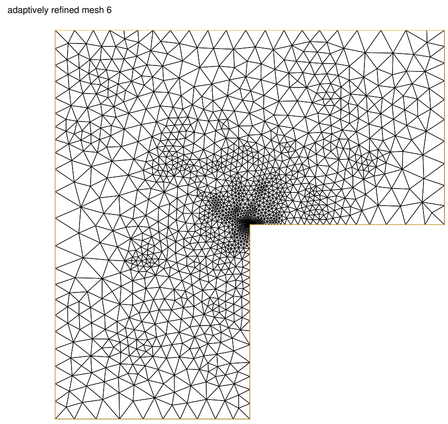
   The above data shows the convergence rate for FEM with a "P1" basis under the adaptively generated meshes is $\mathcal{O}(\text{DoFs}^{-\frac{1}{2}})$, which is optimal in terms of DoFs. Please note that order of convergence in terms of the mesh size h no more has any meaning, since the mesh size is nonuniform and, moreover, the maximal mesh size may not even tend to zero. The effectivity index in turn converges to 1.059.



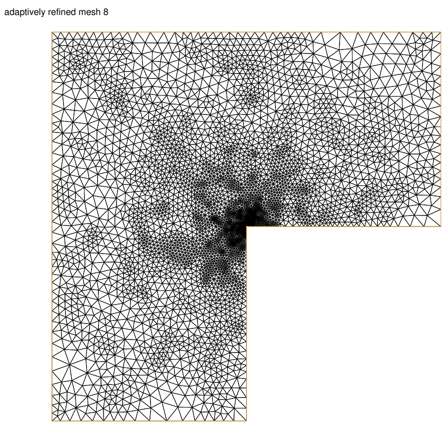


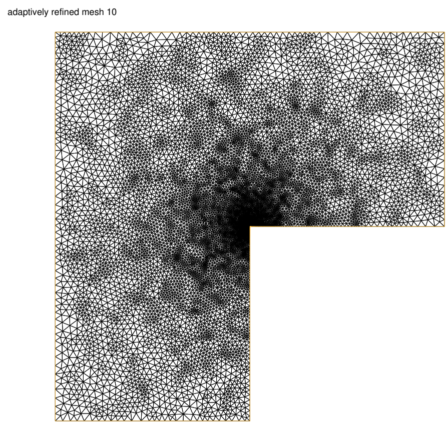
Initial mesh (left) and level 2 adapted mesh (right)





Level 4 adapted mesh (left) and level 6 adapted mesh (right)





Level 8 adapted mesh (left) and level 10 adapted mesh (right)

Figure 9: Adaptively generated meshes sequence for “P1” FEM

4. One should set up the “P2” FE approximation and the “RT2” equilibrated flux and then run the code again. You should see the following data in the command window:

```
Convergence rate 9

1.101054039 2.758097614 13.04280102 -1.965159434 0.8713541543

0.764431557 1.425399035 1.087060819 1.154805182
```

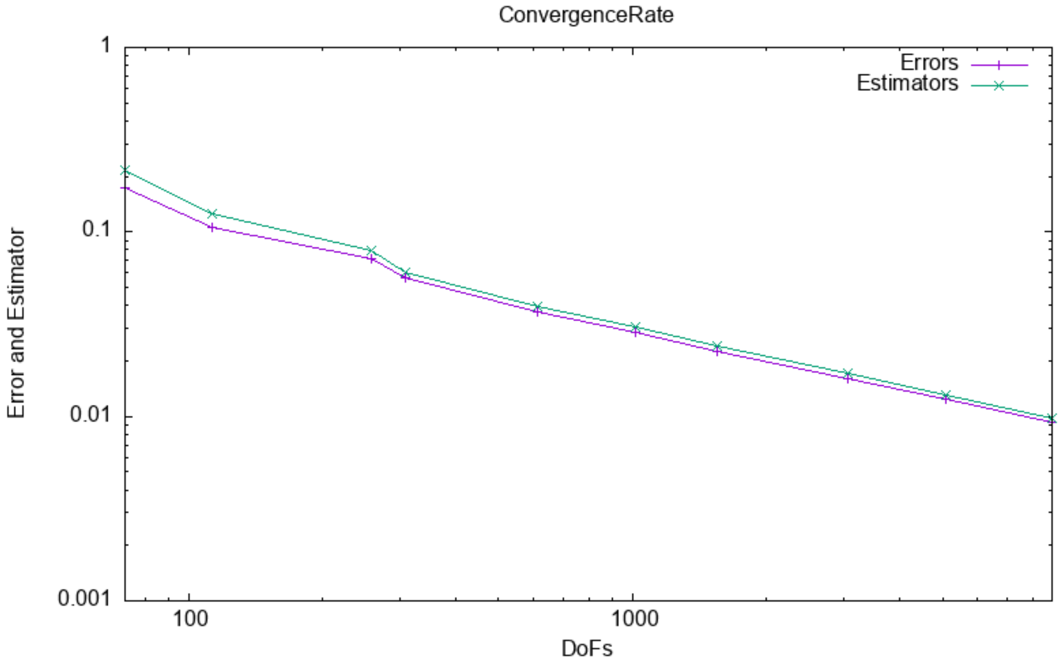


Figure 10: Convergence of the energy error and estimator under adaptively generated meshes with P1 basis

```
Effectivity Index 10
1.387788749 1.419661846 1.299845938 1.173636682 1.084313961
1.060747483 1.038959759 1.034408075 1.023007578 1.022596182
```

The above data shows the convergence rate for FEM with a “P2” basis under the adaptively generated meshes is $\mathcal{O}(\text{DoFs}^{-1})$, which is optimal in terms of DoFs. (Recall that h loses meaning here.) In addition, the effectivity index converges to 1.02. One can also plot the convergence Figure 8 for the FEM with “P2” basis.

So the above numerical results of the FEM confirm the theoretical results that the convergence rate of FEM for a singular solution is optimal in terms of DoFs with the rate $\mathcal{O}(\text{DoFs}^{-\frac{p}{2}})$ with any order p under the adaptive mesh generation.

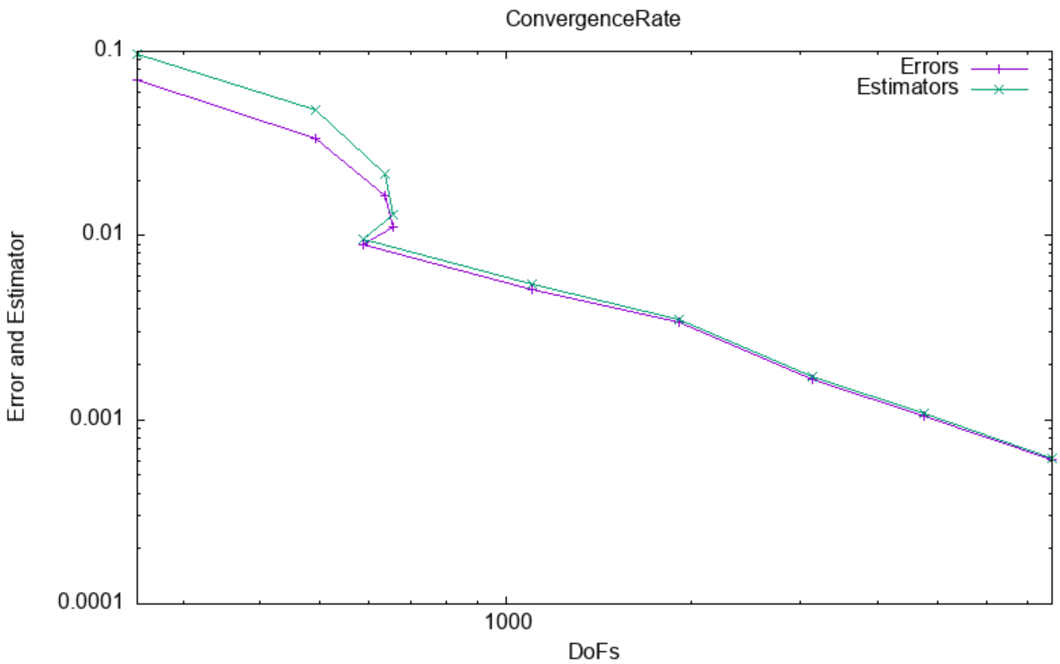


Figure 11: Convergence of the energy error and estimator under adaptively generated meshes with P2 basis