Coding Project 2

Irene Natale - in2582

The code that is presented is thought for solving elliptic partial differential equation of the form

$$\nabla \cdot u = q \quad \Omega$$

$$p = g_D \quad \Gamma_D$$

$$u \cdot n = g_N \quad \gamma_N$$

where $u = -K\nabla p$. The functions q, g_D, g_N can be space varying and the domain is meant to be whatever square shape.
The presented method divides the domain into square elements. It's possible to have different number of nodes on the x and y axis, and this permits to solve problems also for rectangular domains (or square domain problems that need finer refinment on one axis).
The integrals of the finite element method are solved with a Gaussian quadrature discretization. Two dimentions integrals over the area of each element are solved using 4 different integration points ((-1/sqrt(3), -1/sqrt(3)) (1/sqrt(3), -1/sqrt(3)) (1/sqrt(3), 1/sqrt(3)) (-1/sqrt(3), 1/sqrt(3))) with weight functions 1. Linear integral over boundaries are also computed with Gaussian quadrature of the same kind.
The nodes of the mesh, the elements of the mesh are numbered from the bottom left corner to the upper right corner, in anti clockwise order.
The Dirichlet boundary conditions are defined on the upper and lower boundaries of the domain, while the Neumann boundary conditions are defined on the left and right boundaries.
The diffusion coefficient K is defined as a matrix, and it is handled as a matrix along the whole code.

1 Results

The solution plot for the numerical and the real solution is shown in Fig.1 for the case with $64 * 64$ number of nodes. It is possible to notice that the two plots are very similar, and that the method works fine.

Table 1 shows the convergence data. The mesh size is referred to the total number of nodes in the mesh, the number of degree of freedom is equal to the total number of nodes less the nodes on the upper and lower boundary where the Dirichlet boundary conditions are set. Moreover it is possible to oberve the computed L2-error $||e||_{L_2}$ for the various mesh and also the space convergence rate, which is the L2-error over the length of the element side. It is possible to observe that the L2-error decreases as the mesh gets finer, and also the space

Table 1: Convergence table

cicle	mesh size	#cells	#dof	L2-error	space conv rate
1	4*4	9	8	0.0171	0.0684
2	8*8	49	48	0.0079	0.0632
3	16*16	225	224	0.0037	0.0592
4	32*32	961	960	0.0018	0.0576

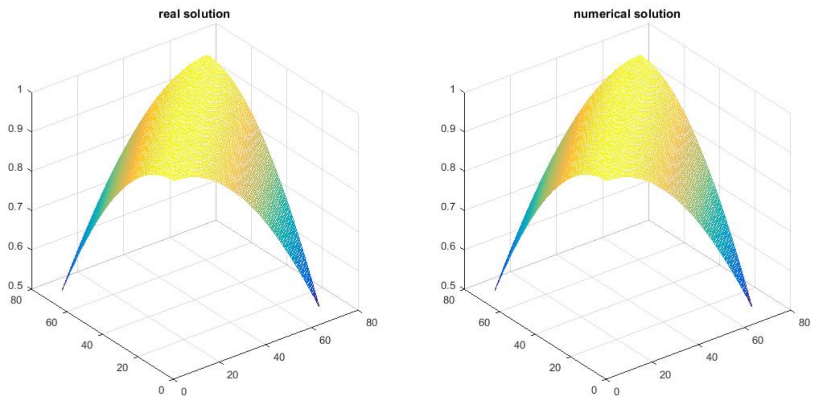

Figure 1: 64x64 mesh solution

convergence rate has the same behaviour.
Fig. 2 and 3 show the L2-error behaviour. in the method basis functions of the type

$$a_0 + a_1x + a_2y + a_3xy = 0$$

are used, so we were expecting a second order convergence of the error. However this is not what we obtain. In fact, it si possible to see from the plots that the error function follows the slope of h and not of h^2. This is probably due to how the boundary conditions are handled, since sometimes a smaller order approximation of the latters can make the global method order decrease.

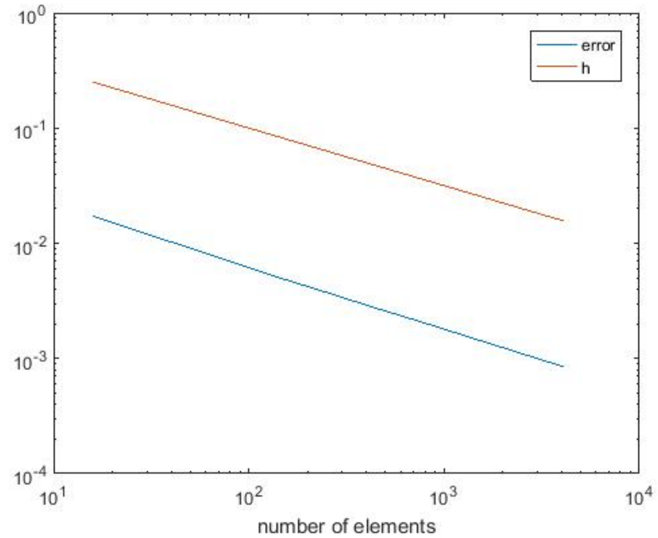

Figure 2

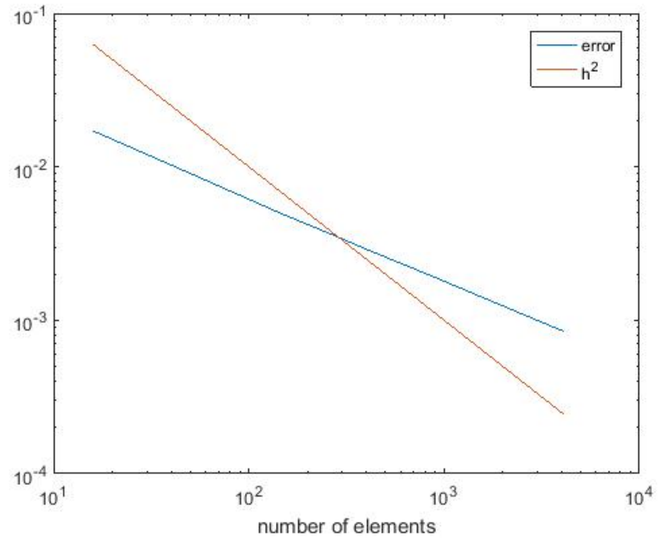

Figure 3