Mandatory assignment 1
MEK4250
ksfd

Christian Pedersen

March 16, 2016

Exercise 1

Mathematical model

The equation set we are to solve;

$$\begin{aligned} -\Delta u &= f \quad \text{in} \quad \Omega = (0,1)^2 \\ u &= 0 \quad \text{on} \quad x = 0, 1 \\ \frac{\partial u}{\partial n} &= 0 \quad \text{on} \quad y = 0, 1 \end{aligned} \tag{1}$$

We are to assume that $u = sin(k\pi x)cos(l\pi y)$. This gives us that

$$f = -sin(k\pi x)cos(l\pi y)\left((k\pi)^2 + (l\pi)^2\right) \tag{2}$$

task a

The general form for a Hilbert norm of dimension p looks like

$$||u||_{H^p} = \left(\int_\Omega \left((\mathbf{u})^2 + \left(\frac{\partial \mathbf{u}}{\partial \mathbf{x}} \right)^2 + \left(\frac{\partial^2 \mathbf{u}}{\partial \mathbf{x}^2} \right)^2 + \left(\frac{\partial^3 \mathbf{u}}{\partial \mathbf{x}^3} \right)^2 + \cdots + \left(\frac{\partial^p \mathbf{u}}{\partial \mathbf{x}^p} \right)^2 \right) d\mathbf{x} \right)^{\frac{1}{2}} \tag{3}$$

The first element in the norm is simply the integral of $\mathbf{u}$ over the domain which is

$$\int_\Omega (\mathbf{u})^2 \, d\mathbf{x} = \frac{1}{4} \tag{4}$$

Using relations

$$\begin{aligned} \int_0^1 sin^2\left(k\pi x\right) dx &= \frac{1}{2} \\ \int_0^1 cos^2\left(k\pi x\right) dx &= \frac{1}{2} \end{aligned} \tag{5}$$

The n'th element in the norm will then be

$$\int_\Omega \left(\frac{\partial^n \mathbf{u}}{\partial \mathbf{x}^n}\right)^2 d\mathbf{x} = \frac{1}{4}\left((k\pi)^2 + (l\pi)^2\right)^n \tag{6}$$

Hence, the H^p norm of $\mathbf{u}$ will then become

$$||u||_{H^p} = \frac{1}{2}\left(\sum_{i=0}^{p}\left((k\pi)^2 + (l\pi)^2\right)^i\right)^{\frac{1}{2}} \tag{7}$$

task b

The L2 and H1 norms are calculated using FEniCS's norm function. When the frequency, k and l, is higher than the number of elements in the mesh we cannot expect to get any good or reliable results. I have therefore chosen to present the results for $k = l = 1$, $k = l = 10$ and when $k = 1$ and $l = 10$ in figures 1, 2 and 3. The rest of the results can be viewed as terminal outputs in the appendix.

We can observe a result that was expected. The L2 error gives a convergence rate around 2 for first order polynomials and a convergence rate around 3 for second order polynomials. The H1 error gives a convergence rate around 1 and 2 for first and second order polynomials.

$k = l = 1$

Polynomial degree	N	L2 error norm	H1 error norm
1	8	0.03277	0.43659
	16	0.00846	0.21816
	32	0.00213	0.10905
	64	0.00053	0.05452
Convergence rate		1.99700	1.00010
2	8	0.00056	0.03318
	16	6.93e-05	0.00838
	32	8.61e-06	0.00210
	64	1.08e-06	0.00052
Convergence rate		3.00179	1.99785

Figure 1: H1 and L2 errors calculated for first and second order polynomials with $k = l = 1$.

$k = 1, l = 10$

Polynomial degree	N	L2 error norm	H1 error norm
1	8	0.67979	16.1499
	16	0.24528	9.17927
	32	0.07865	4.62356
	64	0.02091	2.28339
Convergence rate		1.91122	1.01782
2	8	0.32986	9.01115
	16	0.02530	2.20724
	32	0.00288	0.57230
	64	0.00035	0.14469
Convergence rate		3.04263	1.98377

Figure 2: H1 and L2 errors calculated for first and second order polynomials with $k = 1$ and $l = 10$. We can see that the error is large when the frequency is bigger than number of elements.

task c

A general system

$$||u - u_h||_q \leq Ch^{\alpha}||u||_{p+1} \tag{8}$$

can be solved with the least squares method. Taking the logarithm we

$k = l = 10$

Polynomial degree	N	L2 error norm	H1 error norm
1	8	0.66705	26.4815
	16	0.36553	17.5464
	32	0.17819	10.6024
	64	0.05490	5.43986
Convergence rate		1.69844	0.96274
2	8	0.43561	19.1245
	16	0.08959	6.92035
	32	0.01020	1.97796
	64	0.00113	0.51841
Convergence rate		3.16235	1.93183

Figure 3: H1 and L2 errors calculated for first and second order polynomials with $k = l = 10$. We can see that the error is large when the frequency is bigger than number of elements.

can rewrite the equation as an equation for a straight line

$$log\left(\frac{||u - u_h||_q}{||u||_{p+1}}\right) = \alpha log\,(h) + log(C) \tag{9}$$

where α and $log(C)$ are the two unknown. Choosing q and p we run this equation for multiple h. We can then use the least squares method to approximate a function for the values we have accumulated.

The updated assignment have combined the function norm and the C. This gives us an updated equation

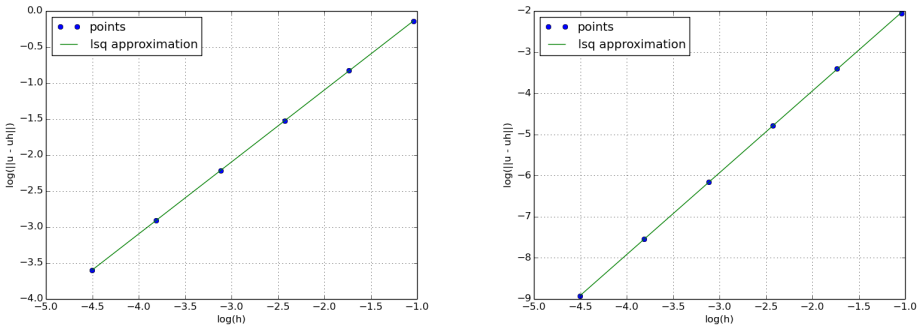
$$log\,(||u - u_h||_q) = \alpha log(h) + log(C_\alpha) \tag{10}$$

When calculating this for an $H1$ and $L2$ error norm we have that q is respectively 1 and 0. Combing the function norm and C should result in a growth in the value of C_α as q increases. For α, I should get a number that resembles the polynomial degree for the H1 norm. The results can be seen in figure 4. For the the L2 norm I should get a number that is the polynomial degree plus one.

The approximated functions can be seen in figures 5 and 6.

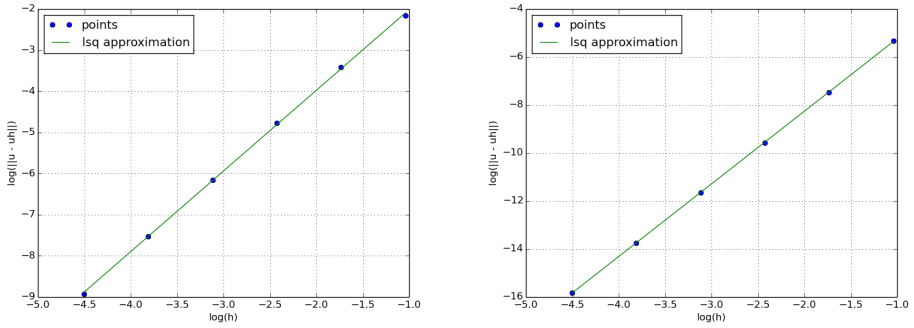
	H1 error norm		L2 error norm	
Polynomial degree	α	C_α	β	C_β
1	0.99944	2.46282	1.96048	0.94583
2	1.98740	1.03005	3.02662	0.10999

Figure 4: List of calculated α, C_α, β and C_β using the least squares method.



(a) Polynomial degree 1 (b) Polynomial degree 2

Figure 5: Figure displays the points from where I find a least squares approximation and the approximation using a H1 error norm.



(a) Polynomial degree 1 (b) Polynomial degree 2

Figure 6: Figure displays the points from where I find a least squares approximation and the approximation using a L2 error norm.

Exercise 2

Mathematical model

The equation set to handle

$$\begin{aligned} -\mu\Delta u + u_x &= 0 \qquad \text{in} \quad \Omega = (0,1)^2 \\ u &= 0 \qquad \text{on} \quad x = 0 \\ u &= 1 \qquad \text{on} \quad x = 1 \\ \frac{\partial u}{\partial n} &= 0 \qquad \text{on} \quad y = 0, 1 \end{aligned} \tag{11}$$

Task a

I assume the function $u(x, y)$ can be written as a product of two seperate function who is each a function of one variable

$$u(x, y) = X(x)Y(y) \tag{12}$$

a method that is known as seperation of variables. I can then rewrite the equation as

$$\frac{X''(x)}{X(x)} - \frac{1}{\mu}\frac{X'(x)}{X(x)} = \frac{Y''(y)}{Y(y)} = \lambda \tag{13}$$

where λ is a unknown constant.

Rewriting the equation I get an equation set

$$\begin{aligned} X''(x) - \frac{1}{\mu}X'(x) - X(x)\lambda &= 0 \\ Y''(y) - Y(y)\lambda &= 0 \end{aligned} \tag{14}$$

Solving for $Y(y)$ first I get

$$Y(y) = c_1 cos\,(\lambda y) + c_2 \sin(\lambda y) \tag{15}$$

Invoking the neumann boundary conditions I end up with

$$Y(y) = c_1 cos\,(\lambda y) \tag{16}$$

where $\lambda = n\pi$ for $n = 0, 1, ..., k$. Since all of these combinations give $c_1 cos\,(\lambda y) = c_1$ I can rewrite my solution as

$$Y(y) = c_1 \tag{17}$$

Choosing $\lambda = 0$ I move on to the $X(x)$ and get

$$X(x) = -b_1\mu + b_2 e^{\frac{1}{\mu}x} \tag{18}$$

Invoking the Dirichlet boundary conditions I get

$$X(x) = \frac{e^{\frac{1}{\mu}x} - 1}{e^{\frac{1}{\mu}} - 1} \tag{19}$$

Combining the two equations give leaves me only one choice for the value of c_1. In order to fulfil the boundary conditions it has got to be 1. Hence the analytical solution is

$$u(x, y) = \frac{e^{\frac{1}{\mu}x} - 1}{e^{\frac{1}{\mu}} - 1} \tag{20}$$

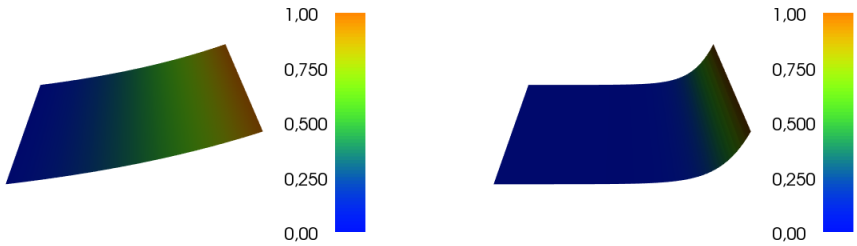
Task b

When calculating the numerical error I ran into machine trouble when setting $\mu < 0.002$. Hence I have chosen the values as displayed in figure 7

	Numerical error				
μ \ N	8	16	32	64	Convergence rate
1.0	1.40e-03	3.51e-04	8.77e-05	2.19e-05	1.99996
0.1	2.38e-02	6.18e-03	1.56e-03	3.91e-04	1.99581
0.01	2.38e-01	1.04e-01	3.82e-02	1.13e-02	1.76192
0.002	7.59e-01	2.83e-01	1.32e-01	6.15e-02	1.09922

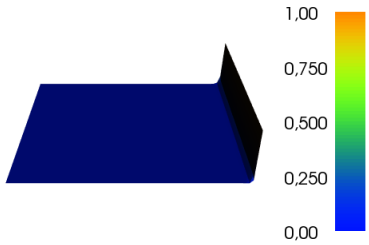
Figure 7: List of calculated numerical error and convergence rate.

We can see from the errors that it seems that something happens to the numerical solution when $\mu \to 0$. This is confirmed when we look at the plots of the numerical solution in figure 8

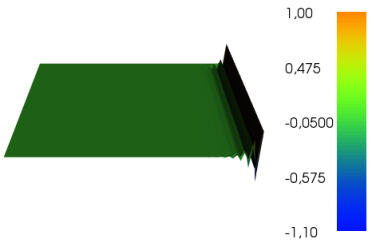


(a) $\mu = 1.0$

(b) $\mu = 0.1$



(c) $\mu = 0.01$



(d) $\mu = 0.002$

Figure 8: Figures display the numerical solution of the equation set. We see that the solution blows up as $\mu \to 0$. For the plots I have used 64 elements in both directions.

Task c

We see that for large μ the error estimates still holds. But as $\mu \to 0$ we see that it does not hold which was expected from task b. A physical interpretation would be that when $\mu \to 0$ we loose the diffusion term and end up with only convection. This results in non-physical oscillations. The results can be seen in figures 9 and 10.

First degree polynomials

	H1 error norm		L2 error norm	
μ	α	C_α	β	C_β
1.0	0.99970	0.21209	1.99955	0.04482
0.1	0.95718	3.89174	1.95515	0.67888
0.01	0.46806	16.83112	1.52306	3.54990
0.002	0.22308	33.24712	1.30163	8.04909

Figure 9: List of calculated α, C_α, β and C_β using the least squares method for first degree polynomials.

Second degree polynomials

	H1 error norm		L2 error norm	
μ	α	C_α	β	C_β
1.0	1.99235	0.01879	2.99325	0.00205
0.1	1.92426	3.13531	2.91130	0.32586
0.01	1.06611	30.94978	2.03929	2.90531
0.002	0.37250	33.44640	1.41275	3.71265

Figure 10: List of calculated α, C_α, β and C_β using the least squares method for second degree polynomials.

Task d

Since we loose the diffusion term when $\mu \to 0$ we can compensate by adding some artificial diffusion. One way is to implement it by the Streamline diffusion/Petrov-Galerkin methods. In our standard weak form of the equation we have

$$\mu \int_\Omega \nabla u \nabla v dn + \int_\Omega u_x v dn = 0 \tag{21}$$

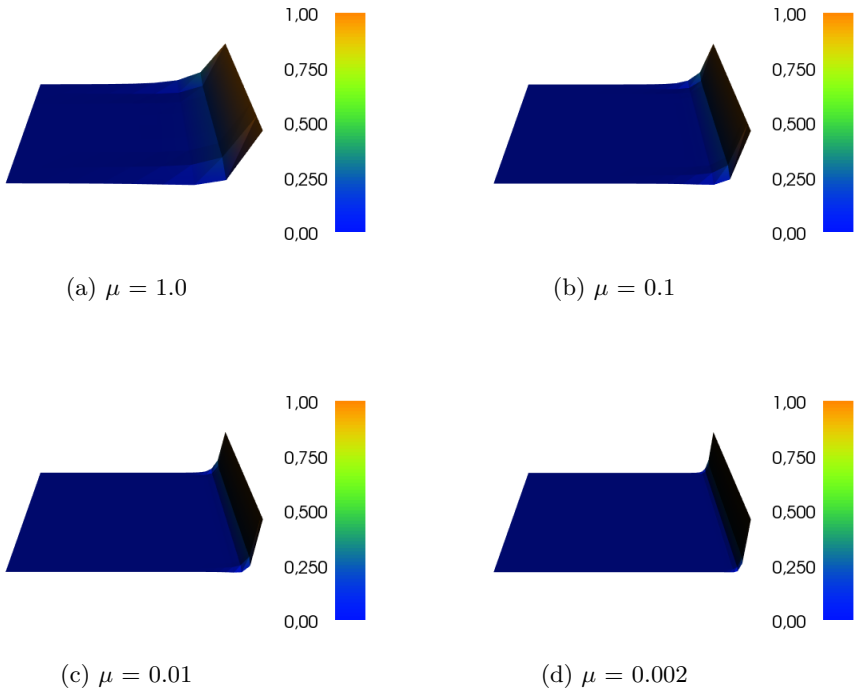
where v is our test function. Now, we define a new test function w

$$w = v + \beta u_x \tag{22}$$

This gives us a new weak form of the equation

$$\mu \int_\Omega \nabla u \nabla w dn + \int_\Omega u_x v dn = \mu \int_\Omega \nabla u \nabla v dn + \int_\Omega u_x v dn + \beta\mu \int_\Omega \nabla u \nabla u_x dn + \beta \int_\Omega u_x^2 dn \tag{23}$$

Implementing this equation in FEniCS we get a smoother function as $\mu \to 0$. This is shown in figure 11.



(a) $\mu = 1.0$ (b) $\mu = 0.1$

(c) $\mu = 0.01$ (d) $\mu = 0.002$

Figure 11: Figures display the numerical solution of the equation set. We see that the solution is smooth as $\mu \to 0$. For the plots I have used 64 elements in both directions.

The numerical error and convergence rate can be seen in figure 12. We see that for the L2 error, the convergence goes from being first order when μ is large and as $\mu \to 0$ the convergence rate goes towards 0.5. This is not

like we did in task b or c. For the H1 norm we also start of with a first order convergence for large μ as we did in task c and as $\mu \to 0$ the convergence rate declines. The tendency is comparable to task c even though we don't get the same numbers.

First degree polynomials in L2

	Numerical error				
μ \ N	8	16	32	64	Convergence rate
1.0	0.00817	0.00396	0.00195	0.00097	1.01033
0.1	0.11679	0.06331	0.03312	0.01698	0.96411
0.01	0.19053	0.12922	0.09505	0.06617	0.80830
0.002	0.18460	0.13204	0.09505	0.06617	0.52244

Figure 12: List of calculated the numerical error and convergence rate.

First degree polynomials in H1

	Numerical error				
μ \ N	8	16	32	64	Convergence rate
1.0	0.04451	0.02257	0.01136	0.00570	0.99480
0.1	1.00675	0.62205	0.35177	0.18819	0.90240
0.01	4.58810	5.39419	4.89193	3.61573	0.43611
0.002	4.68888	6.64172	9.33128	11.78226	-0.33646

Figure 13: List of calculated the numerical error and convergence rate.

Appendix

Terminal output for task 1b;

```
Polynomial degree: 1

k = 1 , l = 1
-----------------
n = 8, L2 errornorm: 0.0327753, H1 errornorm: 0.436592
n = 16, L2 errornorm: 0.00846274, H1 errornorm: 0.218166
n = 32, L2 errornorm: 0.0021332, H1 errornorm: 0.109055
n = 64, L2 errornorm: 0.000534408, H1 errornorm: 0.0545237
Convergence rate:
L2:  [ 1.95341346  1.98810456  1.9970069 ]
H1:  [ 1.00085855  1.00037112  1.00010131]

k = 1 , l = 10
-----------------
n = 8, L2 errornorm: 0.67979, H1 errornorm: 16.1499
n = 16, L2 errornorm: 0.245283, H1 errornorm: 9.17927
n = 32, L2 errornorm: 0.0786529, H1 errornorm: 4.62356
n = 64, L2 errornorm: 0.0209112, H1 errornorm: 2.28339
Convergence rate:
L2:  [ 1.47064012  1.64087592  1.91122526]
H1:  [ 0.8150691   0.9893762   1.01782677]

k = 1 , l = 100
-----------------
n = 8, L2 errornorm: 191.311, H1 errornorm: 2760.52
n = 16, L2 errornorm: 262.037, H1 errornorm: 3506
n = 32, L2 errornorm: 3.07984, H1 errornorm: 281.577
```

```
n = 64, L2 errornorm: 4.68872, H1 errornorm: 433.257
Convergence rate:
L2:  [-0.45385037  6.41077358 -0.60633845]
H1:  [-0.34488573  3.63822299 -0.62169413]

k = 10 , l = 1
-----------------
n = 8, L2 errornorm: 0.679028, H1 errornorm: 16.4037
n = 16, L2 errornorm: 0.240959, H1 errornorm: 9.09787
n = 32, L2 errornorm: 0.0783149, H1 errornorm: 4.60585
n = 64, L2 errornorm: 0.0209202, H1 errornorm: 2.28079
Convergence rate:
L2:  [ 1.49468498  1.6214293   1.90438994]
H1:  [ 0.85042357  0.98206155  1.0139334 ]

k = 10 , l = 10
-----------------
n = 8, L2 errornorm: 0.667057, H1 errornorm: 26.4815
n = 16, L2 errornorm: 0.365538, H1 errornorm: 17.5464
n = 32, L2 errornorm: 0.17819, H1 errornorm: 10.6024
n = 64, L2 errornorm: 0.0549037, H1 errornorm: 5.43986
Convergence rate:
L2:  [ 0.86778813  1.03660406  1.69844236]
H1:  [ 0.59380339  0.72679148  0.96274685]

k = 10 , l = 100
-----------------
n = 8, L2 errornorm: 39.002, H1 errornorm: 1062.03
n = 16, L2 errornorm: 22.1079, H1 errornorm: 871.936
n = 32, L2 errornorm: 2.65025, H1 errornorm: 269.12
n = 64, L2 errornorm: 4.01968, H1 errornorm: 405.333
Convergence rate:
L2:  [ 0.81898752  3.06036154 -0.60095594]
H1:  [ 0.28452461  1.69597022 -0.59085596]

k = 100 , l = 1
-----------------
n = 8, L2 errornorm: 193.305, H1 errornorm: 2769.3
n = 16, L2 errornorm: 262.277, H1 errornorm: 3507.09
n = 32, L2 errornorm: 2.93137, H1 errornorm: 281.67
```

```
n = 64, L2 errornorm: 4.68789, H1 errornorm: 433.7
Convergence rate:
L2:  [-0.44021736  6.48337502 -0.67736392]
H1:  [-0.34075379  3.63819591 -0.62268969]

k = 100 , l = 10
-----------------
n = 8, L2 errornorm: 43.8506, H1 errornorm: 1081.56
n = 16, L2 errornorm: 23.0463, H1 errornorm: 873.593
n = 32, L2 errornorm: 2.4471, H1 errornorm: 266.896
n = 64, L2 errornorm: 4.01373, H1 errornorm: 405.623
Convergence rate:
L2:  [ 0.92805803  3.23539103 -0.71387018]
H1:  [ 0.30807806  1.71068441 -0.60386264]

k = 100 , l = 100
-----------------
n = 8, L2 errornorm: 159.356, H1 errornorm: 3226.29
n = 16, L2 errornorm: 246.862, H1 errornorm: 4686.2
n = 32, L2 errornorm: 2.69686, H1 errornorm: 376.364
n = 64, L2 errornorm: 3.58881, H1 errornorm: 540.467
Convergence rate:
L2:  [-0.6314467   6.51628179 -0.41222695]
H1:  [-0.53854464  3.63821672 -0.52207603]

Polynomial degree: 2

k = 1 , l = 1
-----------------
n = 8, L2 errornorm: 0.000569163, H1 errornorm: 0.0331846
n = 16, L2 errornorm: 6.93424e-05, H1 errornorm: 0.00838941
n = 32, L2 errornorm: 8.61165e-06, H1 errornorm: 0.00210554
n = 64, L2 errornorm: 1.07512e-06, H1 errornorm: 0.00052717
Convergence rate:
L2:  [ 3.03703215  3.00937558  3.00179157]
H1:  [ 1.98387072  1.99437737  1.99785335]

k = 1 , l = 10
-----------------
n = 8, L2 errornorm: 0.329865, H1 errornorm: 9.01115
```

```
n = 16, L2 errornorm: 0.0253099, H1 errornorm: 2.20724
n = 32, L2 errornorm: 0.00288929, H1 errornorm: 0.572304
n = 64, L2 errornorm: 0.000350644, H1 errornorm: 0.144695
Convergence rate:
L2:  [ 3.70410162  3.13091395  3.04263554]
H1:  [ 2.02947042  1.94738672  1.98377136]

k = 1 , l = 100
-----------------
n = 8, L2 errornorm: 289.592, H1 errornorm: 3779.66
n = 16, L2 errornorm: 91.8988, H1 errornorm: 1219.8
n = 32, L2 errornorm: 5.90325, H1 errornorm: 556.318
n = 64, L2 errornorm: 1.80305, H1 errornorm: 186.599
Convergence rate:
L2:  [ 1.65590402  3.96046316  1.71107067]
H1:  [ 1.63161387  1.13266036  1.57596545]

k = 10 , l = 1
-----------------
n = 8, L2 errornorm: 0.331676, H1 errornorm: 8.91002
n = 16, L2 errornorm: 0.025276, H1 errornorm: 2.18345
n = 32, L2 errornorm: 0.00287909, H1 errornorm: 0.568733
n = 64, L2 errornorm: 0.000349565, H1 errornorm: 0.144221
Convergence rate:
L2:  [ 3.713933    3.13408331  3.04198402]
H1:  [ 2.02882047  1.94078637  1.9794685 ]

k = 10 , l = 10
-----------------
n = 8, L2 errornorm: 0.435611, H1 errornorm: 19.1245
n = 16, L2 errornorm: 0.0895993, H1 errornorm: 6.92035
n = 32, L2 errornorm: 0.0102058, H1 errornorm: 1.97796
n = 64, L2 errornorm: 0.00113995, H1 errornorm: 0.518416
Convergence rate:
L2:  [ 2.28147982  3.13409176  3.1623555 ]
H1:  [ 1.46650724  1.80683384  1.93183006]

k = 10 , l = 100
-----------------
n = 8, L2 errornorm: 32.7407, H1 errornorm: 1121.76
```

```
n = 16, L2 errornorm: 8.70218, H1 errornorm: 386.281
n = 32, L2 errornorm: 5.12085, H1 errornorm: 520.797
n = 64, L2 errornorm: 1.57998, H1 errornorm: 178.944
Convergence rate:
L2:  [ 1.91163671  0.76499503  1.69647383]
H1:  [ 1.53804273 -0.43106987  1.54120957]

k = 100 , l = 1
-----------------
n = 8, L2 errornorm: 289.696, H1 errornorm: 3780.34
n = 16, L2 errornorm: 91.9235, H1 errornorm: 1219.86
n = 32, L2 errornorm: 5.90687, H1 errornorm: 556.633
n = 64, L2 errornorm: 1.80292, H1 errornorm: 186.505
Convergence rate:
L2:  [ 1.65603268  3.95996717  1.71206249]
H1:  [ 1.63180096  1.13191775  1.5775134 ]

k = 100 , l = 10
-----------------
n = 8, L2 errornorm: 32.0236, H1 errornorm: 1119.42
n = 16, L2 errornorm: 9.07023, H1 errornorm: 383.828
n = 32, L2 errornorm: 5.12436, H1 errornorm: 521.095
n = 64, L2 errornorm: 1.58037, H1 errornorm: 178.853
Convergence rate:
L2:  [ 1.81992386  0.82376774  1.6971128 ]
H1:  [ 1.54421294 -0.44108434  1.54277102]

k = 100 , l = 100
-----------------
n = 8, L2 errornorm: 293.246, H1 errornorm: 5321.28
n = 16, L2 errornorm: 90.4749, H1 errornorm: 1648.5
n = 32, L2 errornorm: 4.7223, H1 errornorm: 689.092
n = 64, L2 errornorm: 1.47096, H1 errornorm: 288.597
Convergence rate:
L2:  [ 1.69652241  4.25995492  1.68272991]
H1:  [ 1.69062298  1.25838121  1.25564027]
```