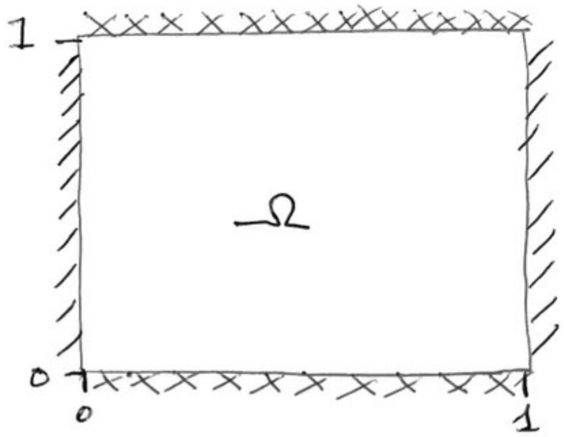
Numerics Exercise 3

i)

//// = Dirichlet boundary conditions on walls $\delta\Omega_D$

XXX = Neumann boundary conditions on walls $\delta\Omega_N$



Dirichlet BC: $u = 0$ on $\delta\Omega_D$

Neumann BC: $\frac{\partial u}{\partial y} = 0$ on $\delta\Omega_N$

Poisson equation: $-\nabla^2 u = f(x,y)$

Domain: $\Omega \in [0,1]^2$

Imposed force: $f(x,y) = 2\pi^2 \sin(\pi x)\cos(\pi y)$

Exact solution: $u_e(x,y) = \sin(\pi x)\cos(\pi y)$

To verify this:

$$\nabla^2 u_e = \partial_x^2 (\sin(\pi x)\cos(\pi y)) + \partial_y^2 (\sin(\pi x)\cos(\pi y))$$

$$= -\pi^2 u_e - \pi^2 u_e$$

$$= -2\pi^2 u_e$$

$$= -2\pi^2 \sin(\pi x)\cos(\pi y) \text{ as expected}$$

$$= -f$$

Ritz Gallerkin principle:

for the energy functional: $\overset{J[u]}{\cancel{I[u]}} = \iint_\Omega \frac{1}{2}(\nabla u)^2 - f(x,y)\,u \, dx\,dy$

we find our solution when $\delta J[u] = J[u + \varepsilon\delta u] - J[u] = 0$

ie when the ~~potential~~ energy of the system is minimised

1 continued)

$$\delta J[u] = J[u + \cancel{\varepsilon\varepsilon}\, \varepsilon \delta u] - J[u]$$

$$= \iint_\Omega \tfrac{1}{2}\left(\nabla(u+\delta u)\right)^2 - f(x,y)\,(\cancel{u}+\delta u)\,dx\,dy$$

$$- \left(\iint_\Omega \tfrac{1}{2}(\nabla u)^2 - \cancel{f(x,y)u}\; dx\,dy\right.$$

$$= \iint_\Omega \frac{1}{2}\left(\cancel{(\nabla u)^2} + 2\nabla(\varepsilon\delta u)\cdot\nabla u + \varepsilon^2(\nabla\delta u)^2 - \cancel{(\nabla u)^2}\right) - \varepsilon f\,\delta u\,dx\,dy$$

$$= \iint_\Omega \frac{\cancel{2}\varepsilon\nabla\delta u\cdot\nabla u}{\cancel{2}} + \frac{\varepsilon^2(\nabla(\delta u))^2}{2} - \varepsilon f\delta u\,dx\,dy$$

$$\frac{\delta J[u]}{d\varepsilon} = \iint_\Omega \frac{\cancel{\nabla(\delta u)\nabla u}}{2}\;\nabla(\delta u)\cdot\nabla u + \varepsilon(\nabla\delta u))^2 - \cancel{\varepsilon} f\delta u\,dx\,dy$$

in $\lim_{\varepsilon\to 0}$

$$= \iint_\Omega \nabla(\delta u)\cdot\nabla u \;\cancel{+} - f\delta u\,dx\,dy$$

∴ we have obtained:

$$\iint_\Omega \nabla(\delta u)\cdot\nabla u - f(x,y)\,\delta u\;dx\,dy = 0$$

Continuous Ritz Gallerkin Formulation

Conditions on variable δu: $\delta u(x,y)$ must be a smooth function within the domain Ω so that $\nabla(\delta u)$ is well defined

δu must also be square integrable ie $\iint_\Omega \delta u(x,y)\,dx\,dy$ must give a finite solution

~~3~~ 1 continued)

Weak Formulation

$-\nabla^2 u = f$ in $\Omega \in [0,1]^2$ $u = 0$ on $\partial\Omega_D$

$\partial\Omega = \partial\Omega_N \cup \partial\Omega_D$ $\nabla u \cdot n = 0$ on $\partial\Omega_N$

~~$-\langle \nabla^2 u, w \rangle = \langle f,$~~

$-\langle \nabla^2 u, v \rangle = \langle f, v \rangle$ where v is chosen as an arbitrary test function that satisfies the boundary conditions

$\langle u, \cancel{w} v \rangle = \iint_\Omega u(x,y)\, v(x,y)\, dx\,dy$ Definition of the inner product

∴ we get:

$$-\iint_\Omega (u_{xx} + u_{yy}) \cancel{w(x}\, v(x,y)\, dx\,dy = \iint_\Omega f(x,y)\, v(x,y)\, dx\,dy$$

LHS: $-\iint_\Omega (u_{xx} + u_{yy})\, v(x,y)\, dx\,dy$

we know: $\partial_x (u_x v) = u_{xx} v + u_x v_x$

$\cancel{\partial_x}\, \partial_y (u_y v) = u_{yy} v + u_y v_y$

rearrange ↓

$u_{yy} v = \partial_y (u_y v) - u_y v_y$

$u_{xx} v = \partial_x (u_x v) - u_x v_x$ } sub into integral

$$-\iint (u_{xx} + u_{yy})\, v\, dx\,dy = -\iint_\Omega \partial_y (u_y v) + \partial_x (u_x v)\, dx\,dy \;\Big] - (A)$$

$$+ \iint_\Omega \cancel{x}(u_y v_y + u_x v_x)\, dx\,dy \;\Big] - (B)$$

1 continued)

to solve Ⓐ: apply green's theorem

$$\int_{\partial\Omega} P\,dx + Q\,dy = \iint_{\Omega} (Q_x \cancel{+} - P_y)\,dydx$$

$$-\iint_{\Omega} \Big[\overbrace{\partial_y (u_y v)}^{-P_y} + \overbrace{\partial_x (u_x v)}^{+Q_x} \Big]\,dxdy = -\int_{\partial\Omega} u_x v\,dy - u_y v\,dx$$

$$= \int_{\partial\Omega_N} u_y v\,dx - \int_{\partial\Omega_D} u_x v\,dy$$

we can choose v (as it is arbitrary) such that

$$\int_{\partial\Omega_D} u_x v\,dy = 0$$

in addition: we know that $u_y = 0$ on $\partial\Omega_N$

∴ we get: Ⓐ $= 0$

∴ our weak formulation is:

$$\iint_{\Omega} [u_y v_y + u_x v_x]\,dxdy = \iint_{\Omega} f(x,y)\,v(x,y)\,dxdy$$

in 2D: $u_y v_y + u_x v_x = \nabla u \cdot \nabla v$

∴ we can write this as:

$$\iint_{\Omega} \nabla u \cdot \nabla v - f v \;dxdy = 0$$

Continuous Weak formulation

where Ω: $0 \le y \le 1$, $0 \le x \le 1$; $f(x,y) = 2\pi^2 \sin(\pi x)\cos(\pi y)$

1 continued)

We can see that the two must be equivalent by ~~looking~~ comparing the ~~the two integrals~~ continuous Ritz Gallerkin formulation and the ~~weak~~ continuous weak formulation.

$$\iint_\Omega \nabla(\delta u) \cdot \nabla u - f(x,y)\,\delta u \, dx\, dy = 0 \qquad \text{RG form}$$

$$\iint_\Omega \nabla v \cdot \nabla u - f(x,y)\, v \, dx\, dy = 0 \qquad \text{Weak form}$$

both must be :
- smooth so that $\nabla v / \nabla(\delta u)$ is well defined
- square integrable so $\iint_\Omega v \, dx dy$ is finite

as v is an arbitrary test function (as is δu) and the two forms share the same integral formulation, it is clear that v ~~as~~ and δu are equivalent. They are both test functions that perturb the system and belong to a specific Hilbert space. specified by the boundary conditions.

2) Finding discrete Ritz Gallerkin Formulation

from before : ~~$J[u] = \iint_\Omega \sum_i \sum_j$~~

NB: u_j and v_j are interchangeable

$$J[u] = \iint_\Omega \tfrac{1}{2} |\nabla u|^2 - f(x,y)\, u \, dx\, dy$$

let: $u \simeq u_n \approx \sum_{j=1}^{N_n} U_j \varphi_j(\bar{x})$

U_j : coefficients

φ_j : basis function

N_n : number of nodes where solution is unknown

$$\iint_\Omega \tfrac{1}{2} \left(\nabla\left(\sum_i u_i \varphi_i\right) \cdot \nabla\left(\sum_j u_j \varphi_j\right)\right) ~~- f u_j~~$$

$$- \sum_j f u_j \varphi_j \, dx\, dy = 0$$

$$\nabla(u_i \varphi_i) = \overset{=0}{\cancel{\varphi_i \nabla u_i}} + u_i \nabla \varphi_i$$

$$= u_i \nabla \varphi_i$$

$N_T = N_n + N_K$

$\uparrow$ total # of nodes

$\leftarrow$ # of known solution nodes (nodes on boundary $\partial\Omega_D$)

2 continued)

$$\therefore \iint_{\Omega} \frac{1}{2}\left(\sum_i \sum_j \cancel{u_i \varphi_i +} u_i \nabla\varphi_i \cdot u_j \nabla\varphi_j\right) - \sum_j u_j f \varphi_j \; dx dy = 0$$

$$\therefore I[u] = \iint_{\Omega} \sum_i \sum_j \frac{u_i \nabla\varphi_i \cdot u_j \nabla\varphi_j}{2} - \sum_j u_j f \varphi_j \; dx dy$$

is the discretised Ritz gallerkin functional

$$= \sum_i \sum_j \frac{A_{ij} u_i u_j}{2} - \sum_j b_j u_j$$

Where we have defined: $A_{ij} = \iint_{\Omega} \nabla\varphi_i \cdot \nabla\varphi_j \; dx dy$

$b_j = \iint_{\Omega} f \varphi_j \; dx dy$

it is convenient not to absorb factor of $1/2$ into A_{ij}

$$\delta I[u] = I[u + \varepsilon \delta u] - I[u]$$

Performing a variation on $\delta I[u]$

here: $\left.\begin{array}{l} u_i \rightarrow u_i + \varepsilon \delta u_i \\ u_j \rightarrow u_j + \varepsilon \delta u_j \end{array}\right\}$ for $I[u + \varepsilon \delta u]$

$$\therefore \delta I[u] = \sum_i \sum_j \frac{A_{ij}}{2} (u_i + \varepsilon \delta u_i)(u_j + \varepsilon \delta u_j) - \sum_j b_j (\cancel{u_j} + \varepsilon \delta u_j)$$

$$- \sum_i \sum_j \frac{A_{ij}}{2} u_i u_j + \cancel{\sum_j b_j u_j}$$

$$= \sum_i \sum_j \frac{A_{ij}}{2} (\cancel{u_i u_j} + u_i \varepsilon \delta u_j + \varepsilon \delta u_i u_j + \varepsilon^2 \delta u_j \delta u_i \cancel{- u_i u_j}) - \sum_j b_j \varepsilon \delta u_j$$

$$= \sum_i \sum_j \frac{A_{ij}}{2} (\varepsilon (u_i \delta u_j + \delta u_i u_j) + \varepsilon^2 \delta u_j \delta u_i) - \sum_j b_j \varepsilon \delta u_j$$

$$\frac{\delta I[u]}{d\varepsilon} = \sum_i \sum_j \frac{A_{ij}}{2} (u_i \delta u_i + \delta u_i u_j + 2\varepsilon \delta u_j \delta u_i) - \sum_j b_j \delta u_j$$

let $\varepsilon \rightarrow 0$

$$= \cancel{\sum_i} \sum_j \sum_i \frac{A_{ij}}{2} (u_i \delta u_j + u_j \delta u_i) - b_j \delta u_j$$

2)

2 continued)

since we defined : $A_{ij} = \iint_{\Omega} \nabla\varphi_i \cdot \nabla\varphi_j \, dx\,dy$

it is clear to see that $A_{ij} = A_{ji}$ (symmetric matrix)

$\therefore$ we can swap $i \longleftrightarrow j$ indices

$$\therefore \frac{\delta I[u]}{d\varepsilon} = \sum_j \sum_i \frac{A_{ij}}{2}\left(u_j \delta u_i + u_j \delta u_i\right) - b_j \delta u_j$$

$$= \sum_j \sum_i A_{ji} u_j \delta u_i - b_j \delta u_j = 0$$

$$\sum_j \sum_i \cancel{A_{ji} u_j \delta u_j} \; A_{ij} u_i \delta u_j - b_j \delta u_j = 0$$

$$\sum_j \sum_i \left(A_{ij} u_i - b_j\right) \cancel{\delta u_j} = 0$$

writing in einstein notation: $\overset{A_{ji}}{\cancel{A_{ij}}} u_i - b_j = 0$

$A_{ij} u_j - b_i = 0$

result when variational principle applied to discretised ritz gallerkin functional

Discretised Weak Formulation

continuous weak form. : $\iint_{\Omega} \nabla u \cdot \nabla v \, dx\,dy = \iint_{\Omega} f v \, dx\,dy$

let: $u(\underline{x}) \simeq u_n(\underline{x}) = \sum_{j=1}^{N_n} u_j \varphi_j$ where φ_j is a square integrable basis function

$\sum_i \sum_j \iint_{\Omega} \nabla(u_j\varphi_j) \cdot \nabla v \; \cancel{d\Omega} - f v \, dx\,dy = 0$ let $v = \varphi_i$ ie we choose our test function to be the same as our basis function

$\sum_i \sum_j \iint_{\Omega} \nabla(u_j\varphi_j) \cdot \nabla\varphi_i - f\varphi_i \, dx\,dy = 0$

this works as long as the basis functions are smooth ~~and~~, square integrable and satisfy the boundary conditions

2 continued)

$$\sum_i \sum_j \iint_\Omega u_j \nabla\varphi_j \cdot \nabla\varphi_i - f\varphi_i \, dx\, dy = 0$$

let us define: $A_{ij} = \iint_\Omega \nabla\varphi_i \cdot \nabla\varphi_j \, d\Omega$ $\qquad d\Omega = dx\,dy$

$$b_i = \iint_\Omega \cancel{fa}\, f\varphi_i \, d\Omega$$

$$\sum_i \sum_j A_{ij} u_j - b_i = 0$$

$\downarrow$

Einstein summation notation: $A_{ij} u_j - b_i = 0$

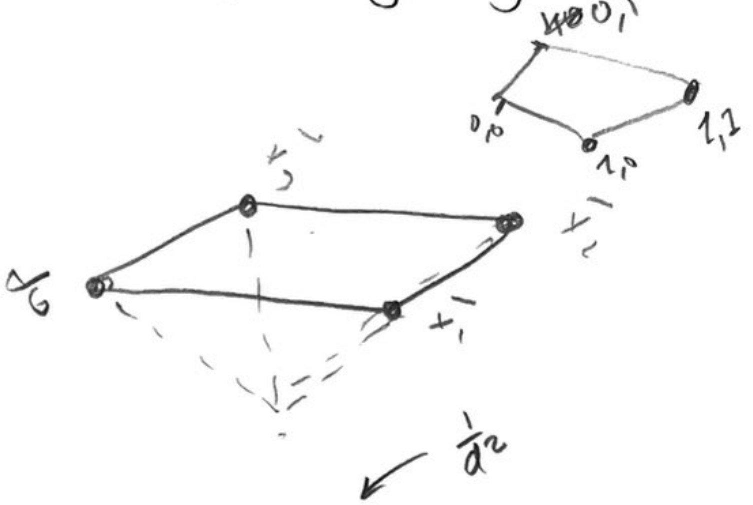
This is the same expression that we ~~obtained~~ from applying the variational principle to the discretised Ritz Gallerkin Formulation,

NB: A_{ij} is not a square matrix as u is known on $\partial\Omega_D$ but not on $\partial\Omega_N$ we can rewrite it in the following way ~~to~~ to turn it into a square matrix by seperating out the Neumann entries (ie those on $\partial\Omega_N$)

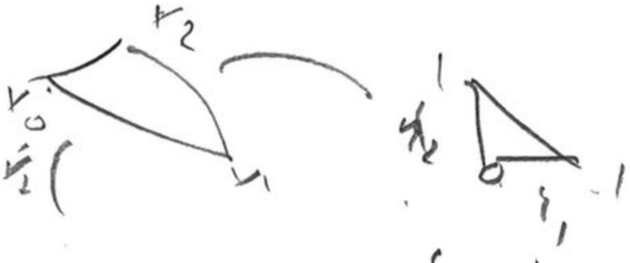
$$A_{ik} u_k - b_i' = - \sum_{m-N_1+1}^{N_T} A_{im} u_m$$

in the case of the poisson equation, I think the RHS disappears?

3) We can use a barycentric coordinate system that expresses the position of a point P within ~~a~~ a rectangular element ~~by the following~~ in the following way



$$(\bar{x}, \bar{y}) = \bar{r} = \frac{1}{4}(1 \pm \xi_1)(1 \pm \xi_2)\vec{x}_i$$

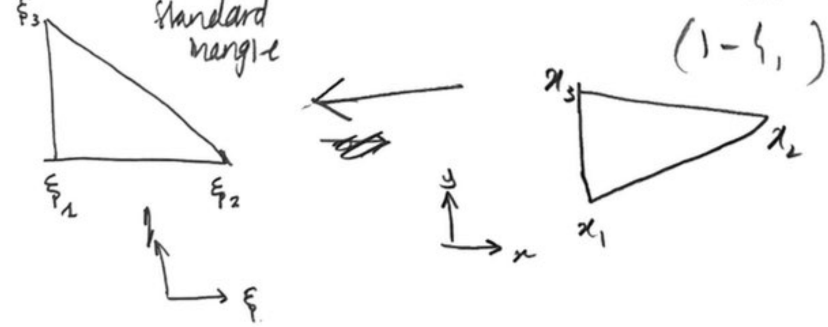


$x_0 \xi_1 + x_1 \xi_2 + x_2 ($

ξ_1, ξ_2

$(1-\xi_1\ )$

you map an irregular triangle / ~~squad~~ to a unit square/ triangle in ~~ξ~~ ξ,η space



general triangle

$x_1 \longrightarrow (0,0) \quad x_2 \longrightarrow (1,0) \quad x_3 \longrightarrow 0,1$

basis functions: $\phi_1(\xi,\eta) = 1 - \xi - \eta$

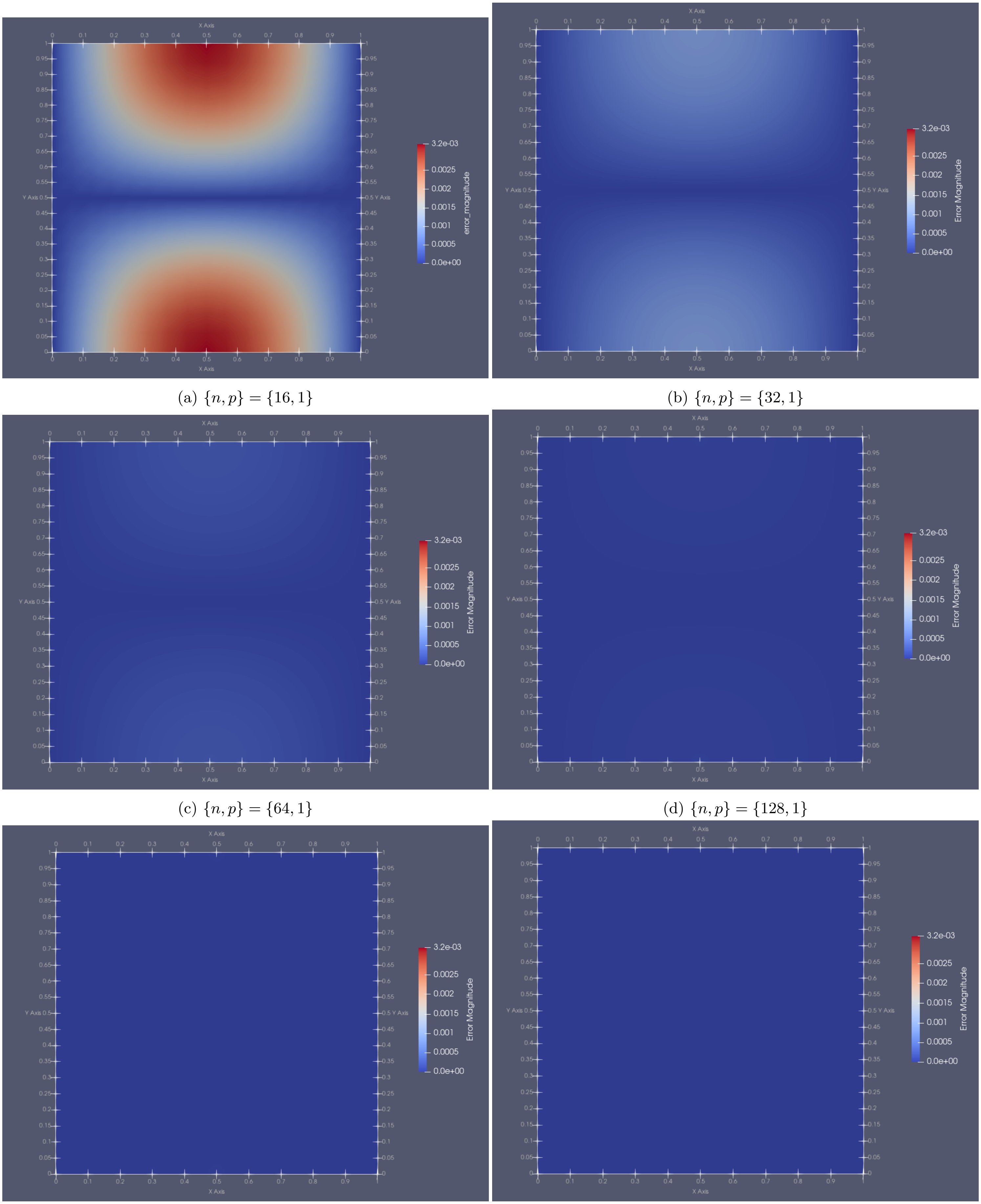
$\phi_2(\xi,\eta) = \xi$

$\phi_3(\xi,\eta) = \eta$

$$x = x_1 + (x_2 - x_1)\xi + (x_3 - x_1)\eta$$

$$y = y_1 + (y_2 - y_1)\xi + (y_3 - y_1)\eta$$

4)



(a) $\{n, p\} = \{16, 1\}$

(b) $\{n, p\} = \{32, 1\}$

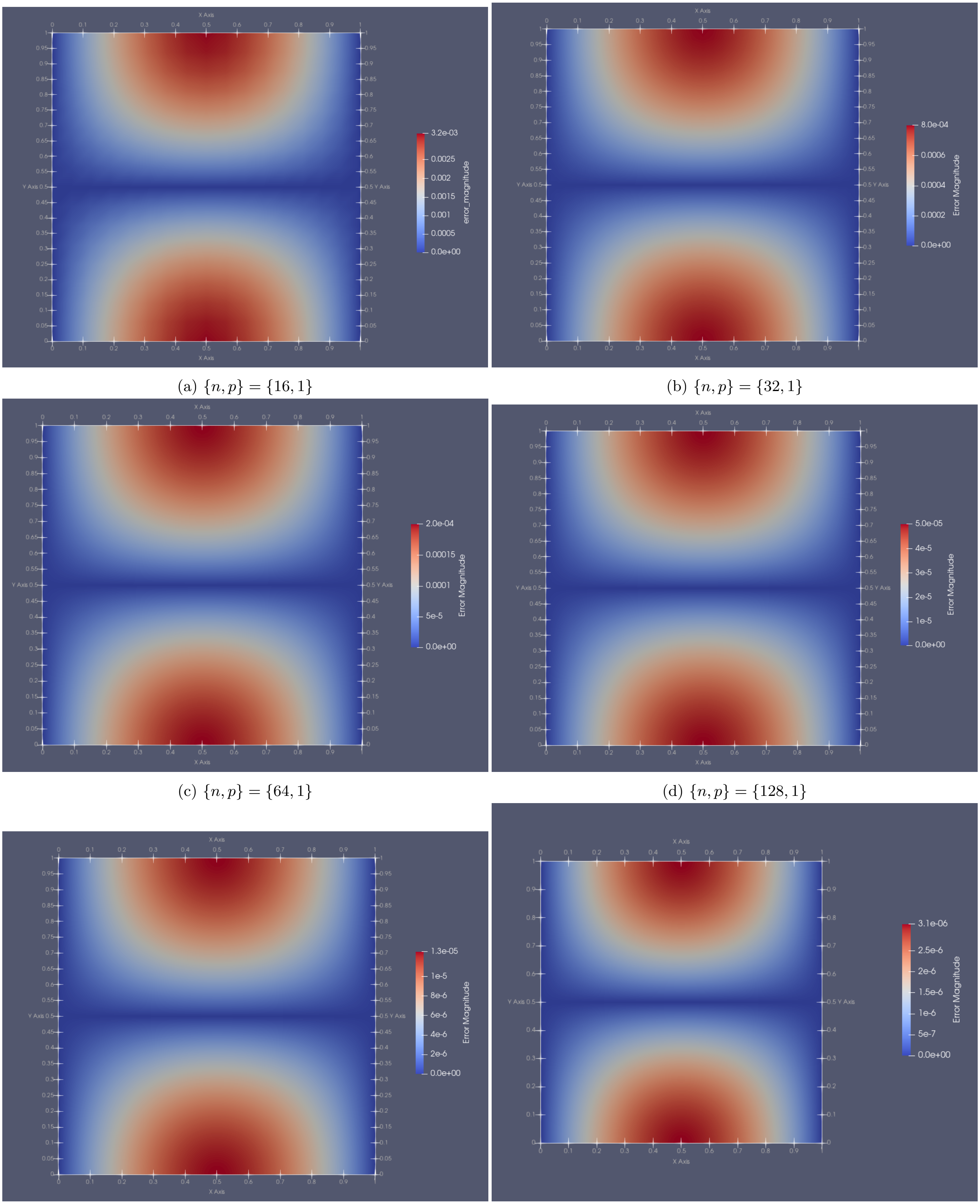
(c) $\{n, p\} = \{64, 1\}$

(d) $\{n, p\} = \{128, 1\}$

(e) $\{n, p\} = \{256, 1\}$

(f) $\{n, p\} = \{512, 1\}$

Figure 1: A figure showing how $\sigma = |u_h - u_e|$ changes as n increases, where u_h is the numerical approximation, u_e is the exact solution and n is the number of nodes within each dimension. The scale on each colour map remains the same, highlighting that the error associated with discretising the Poisson equation using a 1st order FEM is minimised when the resolution of the mesh increases.



(a) $\{n, p\} = \{16, 1\}$

(b) $\{n, p\} = \{32, 1\}$

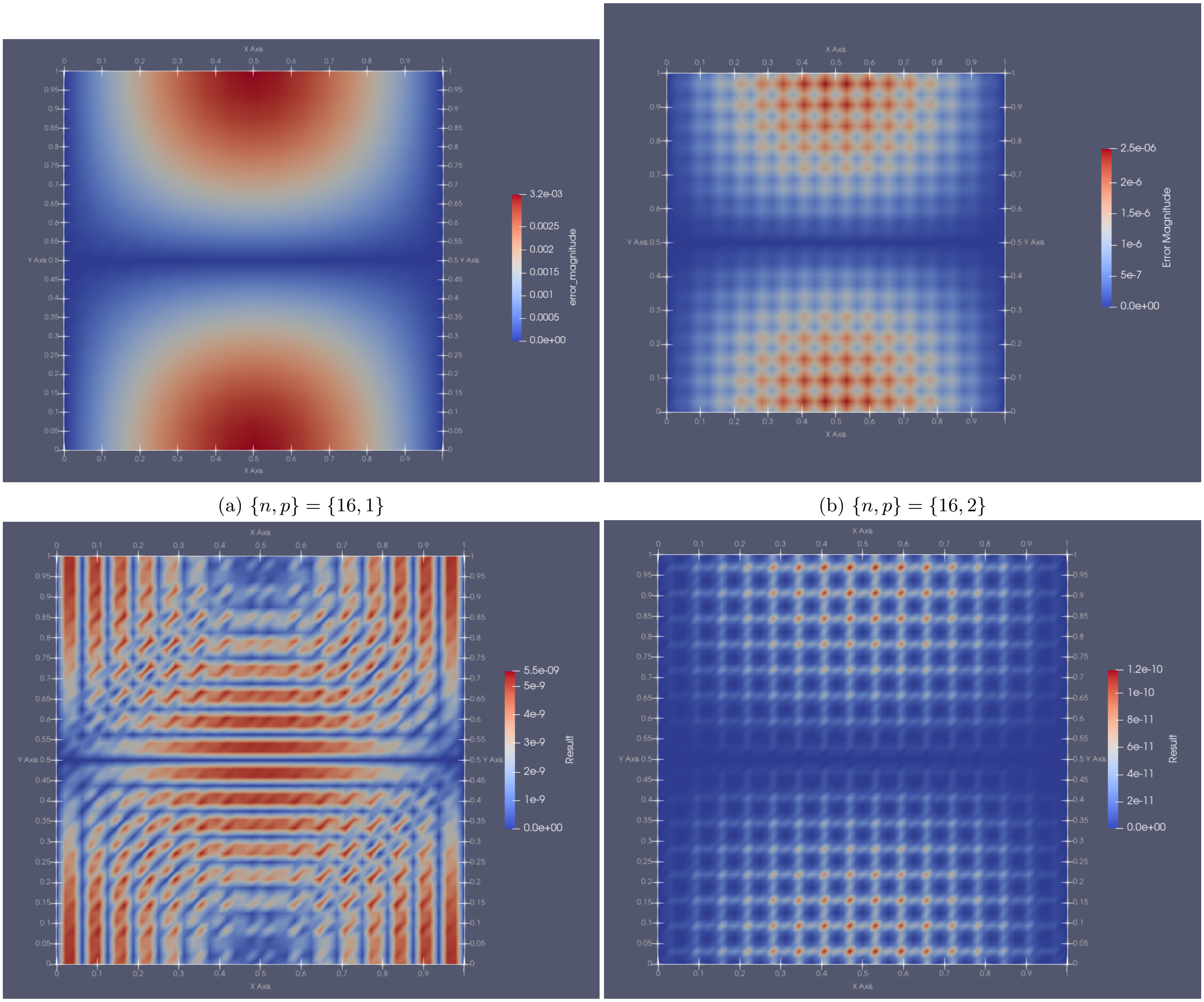
(c) $\{n, p\} = \{64, 1\}$

(d) $\{n, p\} = \{128, 1\}$

(e) $\{n, p\} = \{256, 1\}$

(f) $\{n, p\} = \{512, 1\}$

Figure 2: The same results as in Figure 1, but now each colourbar has been rescaled to show the extremities of $|u_h - u_e$ for the n refinement. By looking at the maxima of each scale, we observe that doubling n decreases the error by a factor of 4. Therefore, we can deduce that $\sigma \propto n^{-2}$ in 2D or (more generally) $|u_h - u_e| \propto n^{-d}$ where d is the number of dimensions.



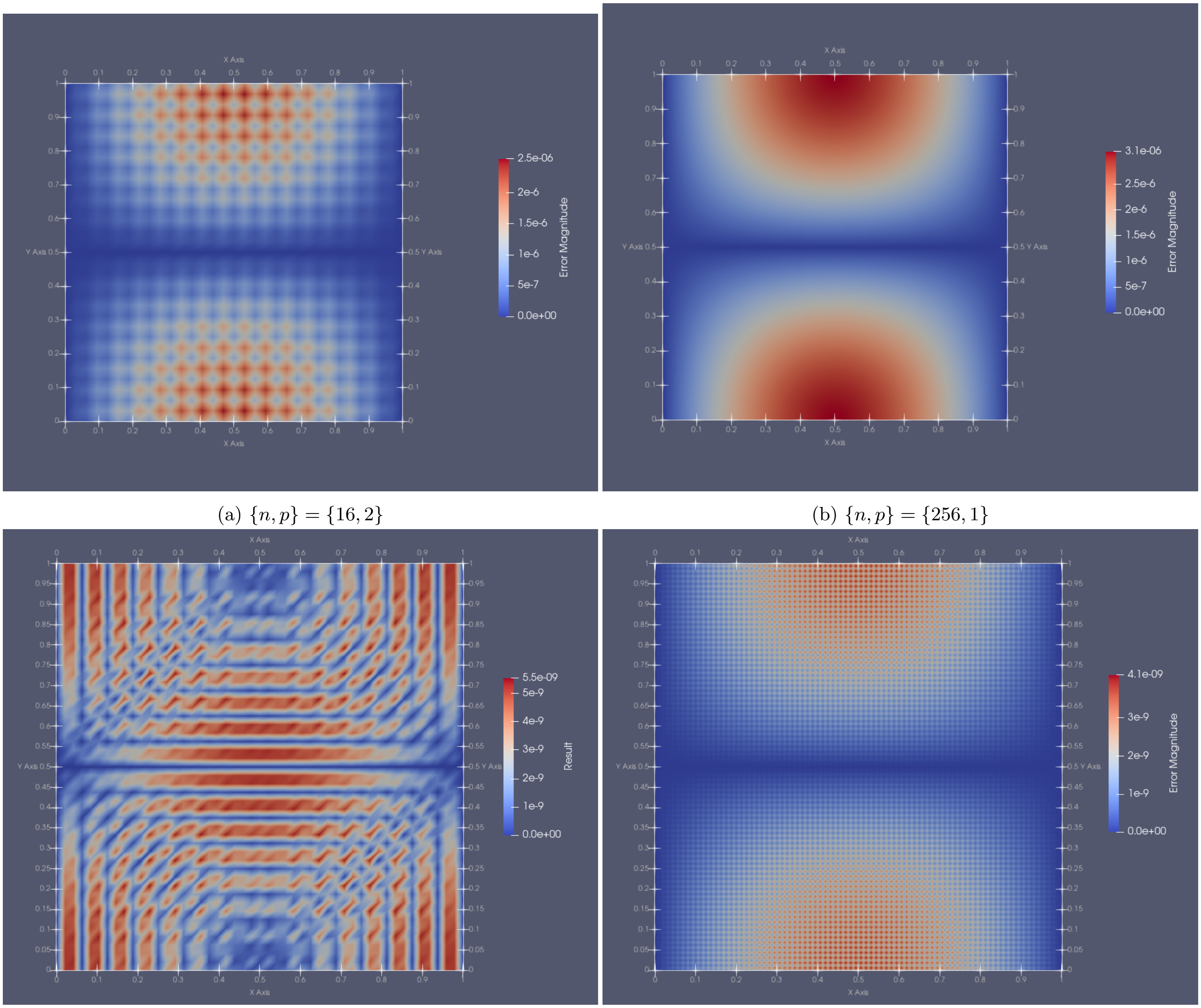
(a) $\{n, p\} = \{16, 1\}$

(b) $\{n, p\} = \{16, 2\}$

(c) $\{n, p\} = \{16, 3\}$

(d) $\{n, p\} = \{16, 1\}$

Figure 3: A p refinement showing the effect of increasing the order p of the Finite Element Approximation when $n = 16$. Notably, increasing p greatly reduces the numerical error σ but does not change the spatial resolution.



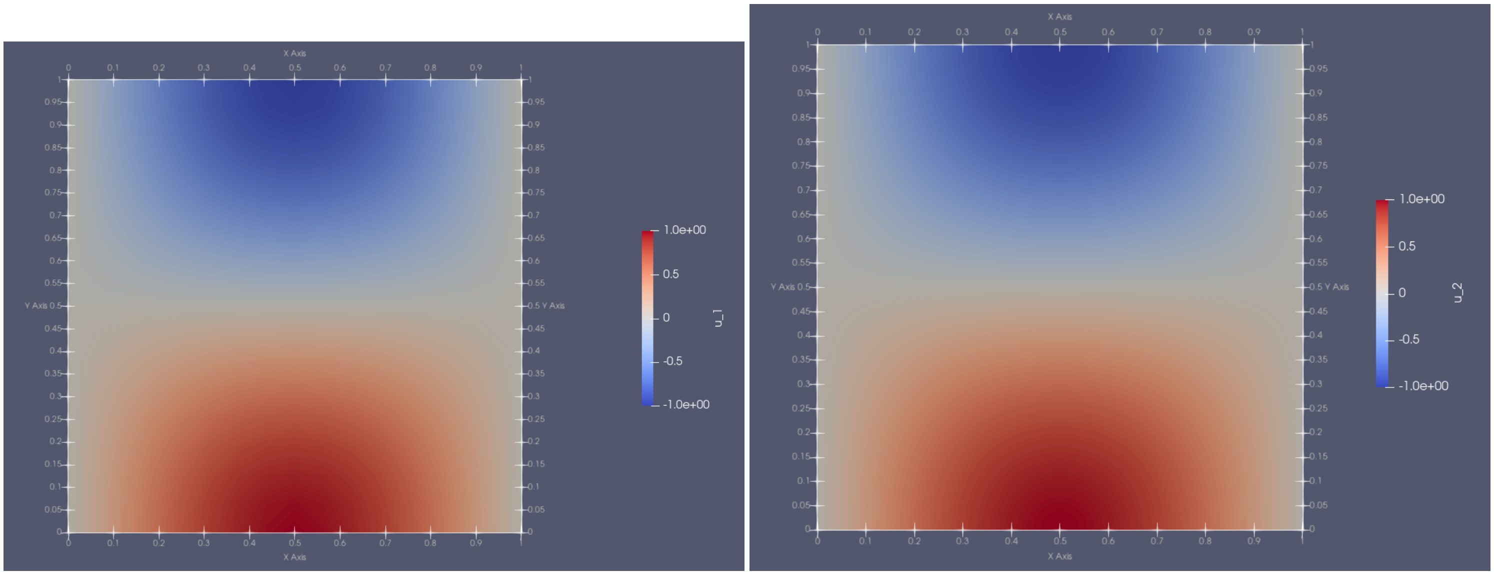
(a) $\{n, p\} = \{16, 2\}$

(b) $\{n, p\} = \{256, 1\}$

(c) $\{n, p\} = \{16, 3\}$

(d) $\{n, p\} = \{80, 2\}$

Figure 4: Two different $\{n, p\}$ pairs: $[(a), (b)]$ and $[(c), (d)]$. This highlights that increasing p is a very effective way of decreasing the error but does not increase the resolution of the mesh.



(a) $\{n, p\} = \{16, 1\}$

(b) $\{n, p\} = \{256, 1\}$

Figure 5: The numerical approximation to the Poisson equation $u_h(x, y)$ plotted over the domain for two different values of n. Since this solution is quite simple, the low spatial resolution of (a) does not significantly affect the results that are obtained within our simulation.

Looking at these results, we can make the following deductions:

- Increasing n (the number of nodes in each dimension) increases the resolution of the graph (which is most easily observed in Figure 4) and the error $\sigma = |u_h - u_e| \propto n^{-2}$ in 2D. We know that $n = \frac{1}{h}$ where h is the side length of each cell within a 2D square mesh. Therefore, an n refinement is the inverse of a h refinement in this case. From this we note that the error scales as $O(h^2)$ when $p = 1$ and the mesh is 2 dimensional. More generally, we can deduce that $\sigma \propto h^{-d}$ for the domain $\{0, 1\}^d$ spanned by cells with side length h and volume h^d.

- p refinement is far more effective than h refinement for minimising the error but it does not increase the spatial resolution of the mesh. Therefore, it is desirable to use a higher value of p to minimise the error but there is a trade-off as you are only minimising the error at a point. Therefore, if h is too low, you do not have sufficient spatial resolution to accurately resolve the flow and you miss important flow features.

- To expand upon this, the convergence scaling which relates p and h is $O(h^p)$ which tells you how the error decreases as h increases. This suggests that there are different scaling laws for different values of p, which makes sense intuitively as a higher order polynomial approximation samples each cell more times (it has a higher degree of freedom). This scaling causes certain $\{h, p\}$ pairs to have a similar amount of numerical error (as observed in Figure 4).

- By comparing Figure 5 with the other figures, we observe that they share the same general profile and symmetry. In particular, the error is maximised where $|u_h(x, y)|$ is large, suggesting that an irregular mesh that places additional cells at these regions could help to resolve the flow more effectively (as an alternative to using a regular mesh and performing refinements of h or p).

In conclusion, h refinement is primarily used as a tool to increase mesh resolution, while p refinement is used to reduce numerical error. Using a mixture of the two is usually the best approach for achieving accurate and highly resolved simulations. Since the two different methods of calculating u_h are mathematically equivalent, we can use either method within this question. I personally opted to use the second method (Ritz-Gallerkin principle) as I think it is more elegant, but you could also use the weak formulation and the results should be equivalent (with any error being attributed to machine precision).

5) I have added some comments at the start of the code that I uploaded to Github in an attempt to translate what Firedrake is doing at each step.

6) I may go back and update my FV solutions over the next week or so (I am still currently deciding whether I want to do this or not).