Análisis Numérico

Segundo Cuatrimestre 2023

TP N° 2 - FEM 2D: Problemas de reacción-difusión singularmente perturbados.

El objetivo es resolver el siguiente problema de reacción-difusión con condiciones de Dirichlet homogeneas:

$$\begin{cases} -\varepsilon^2 \Delta u + u = f & \Omega = [0,1]^2 \\ u = 0 & \partial\Omega. \end{cases}$$

Supongamos que tomamos f tal que

$$u(x,y) = \left(1 - e^{-\left(\frac{1-x}{\varepsilon}\right)}\right)\left(1 - e^{-\left(\frac{1-y}{\varepsilon}\right)}\right)\left(1 - e^{\frac{-x}{\varepsilon}}\right)\left(1 - e^{\frac{-y}{\varepsilon}}\right)$$

es solución (notar que dicha función se anula en el borde de Ω).

Ejercicio 1 Graficar $u(x,y)$ para $\varepsilon = 1$, $\varepsilon = 0.1$, $\varepsilon = 0.01$ y $\varepsilon = 0.001$ utilizando una malla suficientemente fina. Notar el efecto de capa límite hacia los bordes del dominio.
Sugerencia: Para crear la grilla de puntos puede utilizar la función `meshgrid` *de la librería* `numpy`.

Ejercicio 2 Implementar un método de elementos finitos utilizando una malla uniforme de rectángulos y funciones bilineales a trozos, i.e., funciones continuas definidas en Ω, tal que en cada rectángulo es una función de $\mathcal{Q}_1 = \{\phi(x,y) = p(x)q(y), \quad p,q \in \mathcal{P}_1\}$. El problema debe tomar como parametros el valor de ε y N, donde N es la cantidad de subdivisiones del intervalo $[0,1]$. Tomando $\varepsilon = 0.01$ y $\varepsilon = 0.001$, resolver para $N = 8, 16, 32$. Graficar la solución en cada caso.

Ejercicio 3 Una de las mallas más usadas para la resolución de estos problemas de reacción-difusión singularmente perturbados (i.e. $\epsilon << 1$), son las llamadas mallas de Shishkin:

Dado N un número par, definimos $\lambda = \epsilon \ln N$ y generamos una partición de $\Omega = \Omega_{11} \cup \Omega_{12} \cup \Omega_{21} \cup \Omega_{22}$ con

$$\begin{aligned}
\Omega_{11} &= [\lambda, 1-\lambda] \times [\lambda, 1-\lambda], \\
\Omega_{21} &= ([0,\lambda] \cup [[1-\lambda, 1]) \times [\lambda, 1-\lambda], \\
\Omega_{12} &= [\lambda, 1-\lambda] \times ([0,\lambda] \cup [1-\lambda, 1]), \\
\Omega_{22} &= ([0,\lambda] \times ([0,\lambda] \cup [1-\lambda,1])) \cup ([1-\lambda,1] \times ([0,\lambda] \cup [1-\lambda,1])).
\end{aligned}$$

Luego, cada intervalo $[0;\lambda]$ y $[1-\lambda, 1]$ se divide en $N/4$ subintervalos uniformes y los intervalos $[\lambda, 1-\lambda]$ en $N/2$ subintervalos. Esto genera una malla gruesa en $[\lambda, 1-\lambda]$ y una malla fina en $[0,\lambda] \cup [1-\lambda, 1]$ (ver Figura 1).

Resolver nuevamente el problema pero utilizando una malla de Shishkin para los datos del ejercicio 2 y realizar los gráficos correspondientes. ¿Puede notar una mejora en la solución?

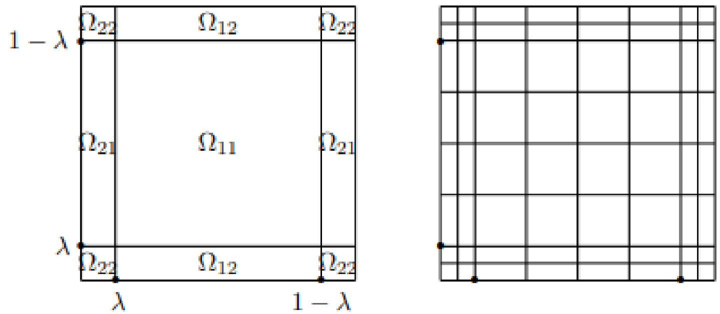

Figure 1: Malla de Shishkin

Ejercicio 4 Tomando las soluciones del problema al utilizar la malla de Shishkin, calcular $\|u - u_h\|_\infty$ para cada caso y estimar el orden (en función de $1/N$) para los valores de ε dados.

Sugerencias:

- Para crear las mallas importar la librería `lib_tp2.py`.
- Para las reglas de cuadratura puede usar el ejercicio 19 de la Práctica 6.
- Para visualizar la solución como una superficie puede utilizar:
  ```
  from mpl_toolkits import mplot3d
  ax = plt.axes(projection='3d')
  ax.plot_surface(x, y, u,cmap='viridis')
  ```
  donde `x,y` estan generados con la función `meshgrid` y `u` es la solución exacta o numérica. Para la solución numérica, dado que para utilizar este graficador es necesario expresar a `u` como una matriz, es conveniente usar el comando `.reshape()`.
- con el comando `ax.view_init(a, b)` puede rotar el gráfico para un mejor analisis.