Control N°1. Difusión no estacionaria con fuente

Enunciado

Sea la ecuación de difusión no estacionaria con fuente expresada en la Eqn. (1)

$$\frac{\partial \phi}{\partial t} - \vec{\nabla} \cdot \left(\nu \vec{\nabla} \phi \right) = \frac{h}{L} \left(\phi_\infty - \phi \right) \tag{1}$$

para el caso unidimensional y con condiciones Dirichlet nulas en ambos extremos siendo su condición inicial $\phi(x, 0) = 0$. Sobre la base de esta ecuación y su solución se solicita:

a) Indicar los fenómenos que se representan con esta ecuación y los problemas físicos que es posible modelar, en particular aquellos que son de su interés profesional y/o científico. Dar una interpretación a la constante h en forma física y mediante los números adimensionales correspondientes.

b) Encontrar su solución analítica por medio del método de Separación de Variables, para ello se sugiere seguir el siguiente proceso,

- Expresando la función solución $\phi(x, t)$ como,

  $$\phi(x, t) = u(x, t) + \tilde{\phi}(x)$$

  donde $\tilde{\phi}(x)$ es la solución del problema estacionario asociado

  $$-\vec{\nabla} \cdot \left(\nu \vec{\nabla} \phi \right) = \frac{h}{L} \left(\phi_\infty - \phi \right)$$

  reemplazarla en la Eq. (1) para hallar un problema simplificado. Verificar las nuevas condiciones de borde e iniciales.

- Utilizando los apuntes del Profesor Daniel J. Arrigo del Departamento de Matemática de la Universidad de Arkansas finalizar la solución.

- Expresar la solución en series de senos. Para computar los coeficientes puede utilizar algún manipulador simbólico como `wxMaxima` o bien el integrador `quad` de `octave`. Para computar estos coeficientes es necesario conocer la expresión de $\tilde{\phi}(x)$ la cual es,

  $$\tilde{\phi}(x) = \phi_\infty \left[1 - \cosh(n\, x) \right] + \phi_\infty \frac{\left[\cosh(n\, L) - 1 \right] \, \mathrm{senh}(n\, x)}{\mathrm{senh}(n\, L)}$$

  donde $n = \sqrt{\frac{h}{L\nu}}$

- Para su control verifique que la solución obtenida satisface la ecuación original así como las condiciones de borde e iniciales. Recordar que las constantes de la serie de Fourier tienen derivada nula.

c) Escribir un script de octave `TranReacDiff(nu, h, phiInf, x, L, t, epsilon, nmax)` que dados los valores de ν, h, L y ϕ_∞ devuelva la solución analítica para un dado tiempo t en cada uno de los puntos de x, evaluando tantos términos de la serie como sea necesario para determinar la solución con un error menor a ϵ y sin superar una cantidad `nmax` de términos.

d) Utilizando el script generado en c) obtener la solución analítica con los siguientes parámetros: `nu = 1, L = 3, phiInf = 10, epsilon = 1E-12, nmax = 100` para `h = 10, 100` en una malla de 10.000 elementos para $t = 0{,}1$. Graficar ambas soluciones junto con las correspondientes soluciones estacionarias en una misma figura e interpretar. El archivo `phiRef.dat` adjunto contiene una solución para $t = 0{,}05$ y `h = 100` con 1.000 elementos para control de la codificación de su script.

e) Programar un resolvedor por volúmenes finitos para la Eqn. (1) y las condiciones de borde e iniciales planteadas tanto para el método Backward Euler como para Crank-Nicolson, utilizando condiciones de borde de segundo orden. Dado el tamaño de la malla se requiere el uso de matrices ralas (sparse) y resolvedores lineales aproximados. Para la generación de la matriz del problema puede entonces hacer (para el caso de un problema de difusión, por ejemplo):

```
K = sparse(n, n);
```

donde n es el número de celdas del problema. Para luego llenar la matriz con sus correspondientes coeficientes mediante un bucle, otra opción es crear la matriz haciendo:

```
% Arreglos de posiciones
rowArray = (2:1:n - 1)';
colArray1 = (1:1:n - 2)';
colArray2 = (2:1:n - 1)';
colArray3 = (3:1:n)';

% Arreglos finales
rowArray = [rowArray; rowArray; rowArray];
colArray = [colArray1; colArray2; colArray3];

% Arreglo de valores
vals = [ones(n - 2, 1)*-1/h*nu; ones(n - 2, 1)* + 2/h*nu;...
    ones(n - 2, 1)*-1/h*nu];

K = sparse(rowArray, colArray, vals, n, n);
```

tras lo cual deberá incorporar las condiciones de borde. De manera similar deberá crear una matriz identidad para finalmente obtener la matriz de avance temporal.

Para la solución del sistema en cada paso de tiempo puede valerse del resolvedor lineal “Conjugate Gradients Squared” de octave de la siguiente manera:

```
[phi, FLAG, RELRES, ITER, RESVEC] = cgs(M, RHS, 1E-9, 1000, 1*u);
```

donde `M` es la matriz de avance temporal y `RHS` es el lado derecho del sistema lineal. Como precondicionador se utiliza el producto `l*u` que puede calcularse fuera del bucle temporal como:

```
% Precondicionador
[l, u, p] = ilu (M, struct ("droptol", 1.e-3));
```

Tenga la precaución de inspeccionar la variable `FLAG` para asegurarse que el resolver ha convergido correctamente en cada paso de tiempo.

f) Utilizando los scripts previamente generados realice un análisis de error de discretización temporal respecto de la solución analítica partiendo de un $\Delta t = 0{,}01$ con $t = 0{,}1$ y realizando cinco refinamientos binarios, tanto para el método Backward Euler (BE) como para Crank-Nicolson (CN). Utilice $h = 10$ y 10.000 celdas. Repetir el caso con 1.000 celdas, detallar lo sucedido e interpretar. El archivo `phiRefNum.dat` adjunto contiene una solución para $t = 0{,}05$, $\Delta t = 0{,}001$ utilizando el método Crank-Nicolson para control de la codificación de su script.

g) Presente sus resultados en un informe de no más de dos páginas, dos columnas, letra 10. Incluir una tabla con las convergencias para BE, CN. Las columnas serán: paso de tiempo, Fo, error L1 BE, orden de convergencia BE, error L1 CN, orden de convergencia CN. Utilizar 4 dígitos significativos. En la primera fila el casillero orden de convergencia quedará vacío (completar con un guión). Presentar una gráfica con lineas de referencia para convergencia de primer orden y segundo orden y allí mismo las gráficas de convergencia para FE y CN. En cada caso dar la discusión correspondiente relacionada a los resultados obtenidos.

h) Entregar junto al informe un archivo comprimido con las rutinas utilizadas tal que ejecutando el archivo principal se presente en pantalla la tabla con el análisis de error solicitado.