Deformación de un cilindro mediante calor

Autores: Alejandro Magnorsky, Andrés Mata Suárez, Mariano Merchante
Instituto Tecnológico de Buenos Aires

Resumen

Palabras clave

I. Introducción

II. Desarrollo

A. Cálculo de la temperatura del cilindro

La temperatura de un cilindro uniforme puede modelarse como una función $u(r,t)$, donde r es la coordenada radial desde el eje del cilindro y t es el tiempo. Dicha función debe satisfacer la siguiente ecuación diferencial:

$$\frac{\partial^2 u}{\partial r^2} + \frac{1}{r}\frac{\partial u}{\partial r} = \frac{1}{4K}\frac{\partial u}{\partial t} \tag{1}$$

para $\frac{1}{2} < r < 1$, $0 < t < 10$ y $K = 0.1$.

Las condiciones de contorno, para $0 \leq t \leq 10$, son:

$$u(\frac{1}{2}, t) = t \tag{2}$$

$$u(1,t) = 100 + 40t \tag{3}$$

Además, la condición inicial, para $\frac{1}{2} \leq r \leq 1$, es:

$$u(r,0) = 200(r - 0.5) \tag{4}$$

El objetivo es, entonces, encontrar una aproximación $v_k^m \approx u(\frac{1}{2} + k\Delta r, m\Delta t)$, donde $k = 0, 1, \ldots, n$ siendo $n = \frac{1-\frac{1}{2}}{\Delta r}$ y $m = 0, 1, \ldots, l$ con $l = \frac{10}{\Delta t}$.

Para ello, se utilizan las siguientes diferencias finitas:

- Centradas

$$\frac{\partial^2 u}{\partial r^2} = \frac{u(r+\Delta r, t) - 2u(r,t) + u(r-\Delta r, t)}{\Delta r^2} + O(\Delta r^2) \tag{5}$$

$$\frac{\partial u}{\partial r} = \frac{u(r+\Delta r, t) - u(r-\Delta r, t)}{2\Delta r} + O(\Delta r^2) \tag{6}$$

- Progresiva

$$\frac{\partial u}{\partial t} = \frac{u(r, t+\Delta t) - u(r,t)}{\Delta t} + O(\Delta t) \tag{7}$$

Escribiendo las aproximaciones de las diferencias centradas en la ecuación (1), se obtiene el siguiente esquema:

$$\frac{v_{k+1}^m - 2v_k^m + v_{k-1}^m}{\Delta r^2} + \frac{1}{\frac{1}{2} + k\Delta r}\frac{v_{k+1}^m - v_{k-1}^m}{2\Delta r} = \frac{1}{4K}\frac{v_k^{m+1} - v_k^m}{\Delta t} \tag{8}$$

$$v_k^{m+1} = v_k^m + \frac{4K\Delta t}{\Delta r^2}(v_{k+1}^m - 2v_k^m + v_{k-1}^m) + \frac{4K\Delta t}{(\frac{1}{2}+k\Delta r)2\Delta r}(v_{k+1}^m - v_{k-1}^m) \tag{9}$$

Las condiciones de contorno y la inicial se deducen de las ecuaciones (2), (3) y (4) :

$$v_0^m = m\Delta t \tag{10}$$

$$v_n^m = 100 + 40m\Delta t \tag{11}$$

$$v_k^0 = 200(\frac{1}{2} + k\Delta r - 0.5) \tag{12}$$

Las diferencias finitas, utilizadas para construir el esquema de la ecuación (9), tienen errores de $O(\Delta r^2)$ y $O(\Delta t)$. Por lo tanto, el esquema tiene $O(\Delta r^2 + \Delta t)$, es decir, orden (2,1).

Para analizar la estabilidad, se aplica la Transformada de Fourier de Tiempo Discreto en ambos miembros del esquema de la ecuación (9), junto a las propiedades de desplazamiento en el tiempo y linealidad (Mathews y Fink, 1992), obteniendo:

$$V_k^{m+1} = V_k^m + \frac{4K\Delta t}{\Delta r^2}(V_k^m e^{i\omega} - 2V_k^m + V_k^m e^{-i\omega}) + \frac{4K\Delta t}{r2\Delta r}(V_k^m e^{i\omega} - V_k^m e^{-i\omega}) \tag{13}$$

Despejando V_k^m:

$$V_k^{m+1} = (1 + \frac{4K\Delta t}{\Delta r^2}(e^{i\omega} + e^{-i\omega}) - 2\frac{4K\Delta t}{\Delta r^2} + \frac{4K\Delta t}{r2\Delta r}(e^{i\omega} - e^{-i\omega}))V_k^m \tag{14}$$

$$V_k^{m+1} = (1 + 2\frac{4K\Delta t}{\Delta r^2}(\cos\omega - 1) + i2\frac{4K\Delta t}{r2\Delta r}\sin\omega)V_k^m \tag{15}$$

Se define $\rho = 1 + 2\frac{4K\Delta t}{\Delta r^2}(\cos\omega - 1) + i2\frac{4K\Delta t}{r2\Delta r}\sin\omega$. Entonces:

$$V_k^{m+1} = \rho V_k^m \tag{16}$$

$$V_k^{m+1} = \rho^{m+1} V_k^0 \tag{17}$$

Por lo tanto, $|\rho| < 1$ para que el método sea estable. Para que eso se cumpla, como mínimo, $|Im(\rho)| < 1$:

$$|2\frac{4K\Delta t}{r2\Delta r}\sin\omega| < 1 \Rightarrow \frac{4K\Delta t}{r2\Delta r} < \frac{1}{2} \Rightarrow \Delta t < \frac{r\Delta r}{4K} \tag{18}$$

Considerando los casos en que $r = \frac{1}{2}$ y $r = 1$, se obtiene que el caso más restrictivo es que $\Delta t < \frac{\Delta r}{8K}$.

En la figura 1, se detalla la implementación de un programa que, dados Δr y Δt, utiliza las condiciones de contorno y la inicial de las ecuaciones (10), (11) y (12) para armar parte de la matriz v que representa a la función v_k^m, donde k es la columna y m es la fila. Para completar el resto, usa el esquema de la ecuación (9).

III. Resultados

IV. Conclusiones

Referencias

Mathews, John H., Fink, Kurtis D., “Numerical Methods Using MATLAB”, Prentice Hall, 1999

Listing 1: Implementación del cálculo de v_k^m.

```
function v = finiteDifferences(deltaR, deltaT)

        K = 0.1;
        n = (1/2)/deltaR;
        l = 10/deltaT;

        v = zeros(l+1,n+1);
        for k = 0:n
                v(1,k+1) = 200*(1/2 + k*deltaR - 0.5);
        endfor
        for m=0:l
                v(m+1,1) = m*deltaT;
                v(m+1,n+1) = 100 + 40*m*deltaT;
        endfor

        for m=1:l
                for k=2:n
                        v(m+1,k) = v(m,k) + 4*K*deltaT/deltaR^2 * (v(m,k+1) - 2*v(m,k) + v(m,k-1)) + 4*K*
                            deltaT/((1/2+(k-1)*deltaR)*2*deltaR) * (v(m,k+1) - v(m,k-1));
                endfor
        endfor

endfunction
```