

Certamen #3 Mat-015 (Etapa 1)

2^{do} Semestre 2014

Victor Gonzalez Rodriguez
victor.gonzalezro@alumnos.usm.cl
2773029-9

10 de diciembre de 2014

$$(\mathbf{P}) \begin{cases} u_t - c^2 u_{xx} = f(x, t) & 0 < x < l, \ c > 0 \\ u(0, t) = u(l, t) = 0 & t > 0 \\ u(x, 0) = g(x) & 0 < x < l \end{cases}$$

1. Planteo de esquema de diferencias finitas.

1.1. Planteo analítico.

Se nos dice que los valores a aproximar son denotados por $u(x_j, t_k) = u_j^k$, y que las derivadas se definen mediante las siguientes discretizaciones:

$$u_t(x_j, t_k) \approx \frac{u_j^k - u_j^{k-1}}{h_t}, \quad u_{xx} \approx \frac{u_{j+1}^k - 2u_j^k + u_{j-1}^k}{h_x^2}$$

Luego, podemos reescribir la EDP de manera discreta:

$$u_t - c^2 u_{xx} = f(x, t) \tag{1}$$

$$u_t = c^2 u_{xx} + f(x, t) \tag{2}$$

$$u_t(x_j, t_k) = c^2 u_{xx}(x_j, t_k) + f(x_j, t_k) \tag{3}$$

$$\frac{u_j^k - u_j^{k-1}}{h_t} = c^2 \frac{(u_{j+1}^k - 2u_j^k + u_{j-1}^k)}{h_x^2} + f_j^k \tag{4}$$

$$u_j^k - u_j^{k-1} = \frac{c^2 h_t}{h_x^2} (u_{j+1}^k - 2u_j^k + u_{j-1}^k) + h_t f_j^k \quad (5)$$

En (5) denotaremos la fracción de constantes como λ : $\lambda = \frac{c^2 h_t}{h_x^2}$. Además, podemos notar que el problema tiene la forma de *atrás en el tiempo*, el cual se resuelve mediante métodos de matrices y de descomposición LU^{1 2}.

Considerando lo anterior, además que el profesor mencionó constantemente que la resolución del problema era en pocas líneas de código, vamos a suponer que la forma que buscamos, es la del esquema progresivo, el cual es mas sencillo de ejecutar.

Luego, nuestro problema queda de la siguiente manera:

$$u_j^{k+1} - u_j^k = \lambda (u_{j+1}^k - 2u_j^k + u_{j-1}^k) + h_t f_j^k \quad (6)$$

$$u_j^{k+1} = u_j^k + \lambda (u_{j+1}^k - 2u_j^k + u_{j-1}^k) + h_t f_j^k \quad (7)$$

Por lo tanto, en (7) tenemos discretizada la solución de la EDP original. Luego, las condiciones de borde se escriben como:

$$\begin{aligned} u(0, t) = 0 &\Rightarrow u(x_0, t_j) = 0 \Rightarrow u_0^j = 0 \\ u(l, t) = 0 &\Rightarrow u(x_{n+1}, t_j) = 0 \Rightarrow u_{n+1}^j = 0 \end{aligned}$$

Y la condición inicial queda dada por: $u(x, 0) = g(x) \Rightarrow u(x_k, t_0) = g(x_k) \Rightarrow u_0^k = g(x_k)$.

Adicionalmente, se dice que $\lambda < 1$, para que el problema se considere estable, y pueda converger.

1.2. Planteo del algoritmo.

- Generar un mallado que contenga las regiones para las cuales esta definido el problema.
- Aplicar la condición inicial $u_0^k = g(x_k)$.
- Iterar sobre el mallado, aplicando condiciones de borde y calculando los u_j^{k+1} .
- Entregar resultado.

¹http://web.cecs.pdx.edu/~gerry/class/ME448/notes/pdf/BTCS_slides.pdf

²http://www.mathworks.com/matlabcentral/newsreader/view_thread/308751

2. Implementación en Matlab.

```
function calor(T,l,dt,dx,c)
    % u(x,t) = e-3tsin(x) + xt
    % f(x,t) = x
    % g(x) = sin(x)

    % mallado
    n      = (T-0)/dt;
    m      = (l-0)/dx;
    lambda = c^2*dt/dx^2      % lambda debe ser menor a 1
    u      = zeros(m+1,n+1);

    x = 0:dx:l;
    t = 0:dt:T;

    % valores iniciales
    for j=1:m+1,
        u(j,1) = sin(x(j));    % u(x,0) = g(x)
    end

    % obtencion de discretizacion
    for k=1:n
        u(1,k+1)=0.;           % u(0,t) = 0
        for j=2:m
            u(j,k+1)=u(j,k)+lambda*(u(j+1,k)-2*u(j,k)+u(j-1,k))+dt*x(j);
        end
        u(m+1,k+1)=0.;         % u(l,t) = 0
    end

    % comparar graficamente con solucion real
    [xx,tt]=meshgrid(x,t);
    exact=exp(-3.*tt).*sin(xx) + xx.*tt;

    figure
    contour3(u)

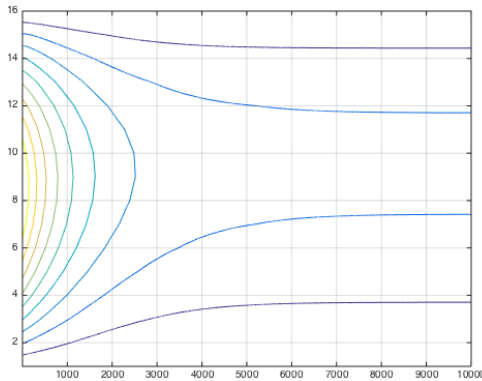
    figure
    contour3(exact)
```

3. Comparación de resultados con solución real.

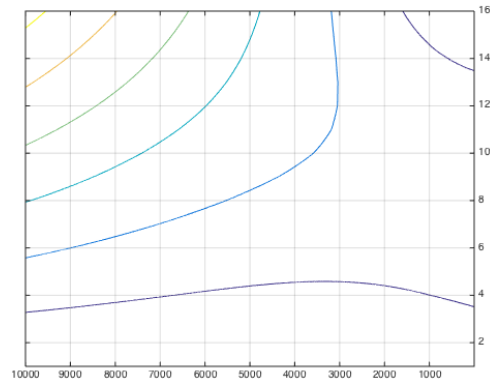
Para el siguiente ejemplo, se utilizaron los siguientes parámetros:

- $u(x, t) = e^{-3t} \sin(x) + xt$
- $f(x, t) = x$
- $g(x) = \sin(x)$
- $t \in [0, 1]$ en intervalos de 0,0001.
- $x \in [0, \pi]$ en intervalos de $\frac{\pi}{15}$.
- $\lambda = 0,0205 < 1$

3.1. Líneas de contorno: vista superior



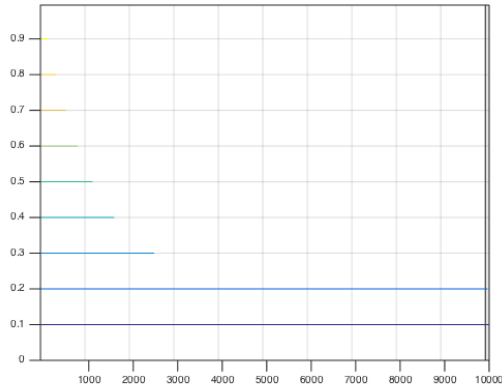
(a) Estimado



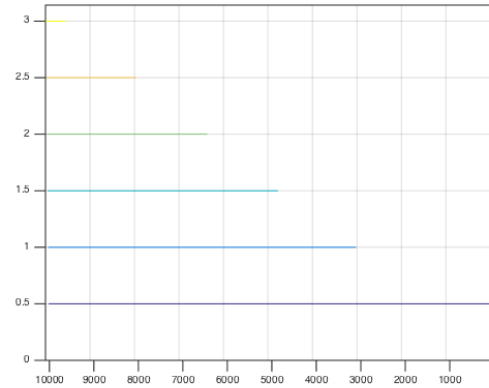
(b) Real

Aquí podemos apreciar claramente, que la estimación mediante diferencias finitas se asemeja a la solución real. Si bien, no son exactamente iguales, el comportamiento es bastante similar.

3.2. Líneas de contorno: vista lateral

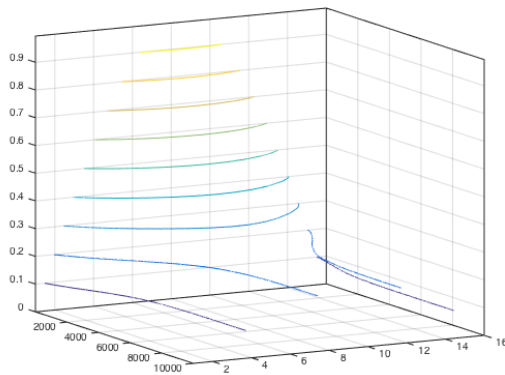


(a) Estimado

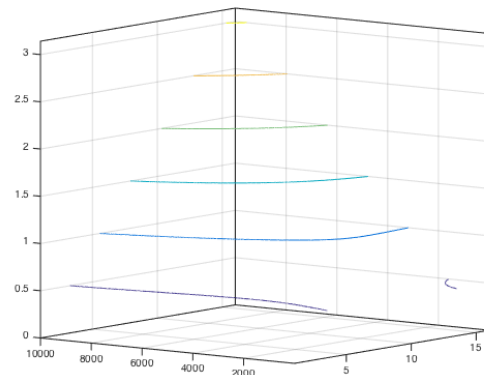


(b) Real

3.3. Líneas de contorno: vista diagonal



(a) Estimado



(b) Real

Por lo tanto, podemos apreciar gráficamente, que la estimación mediante diferencias finitas es una herramienta muy útil, que permite encontrar rápidamente estimaciones de funciones que pueden ser complejas de encontrar en primera instancia.

Las diferencias (error matemático), es una de las grandes desventajas de este método, el cual se arrastra y puede ser incremental, y es por eso que el parámetro λ debe ser menor a 1, para asegurar que el problema se comporte de manera *estable* (aunque esto no asegura que el problema converja totalmente).