

Práctica de Laboratorio:

Solución Aproximada de Ecuaciones Diferenciales Parciales usando Series de Fourier

Fredy Vides

February 19, 2025

1 Introducción

En esta práctica de laboratorio, se estudia la resolución numérica de la ecuación del calor y la ecuación de la onda en el intervalo $0 < x < L$ con condiciones de frontera homogéneas de Neumann. En el caso de la ecuación del calor

$$u_t - ku_{xx} = 0, \quad 0 < x < L, \quad t > 0 \quad (1)$$

con condiciones de frontera homogéneas de Neumann:

$$u_x(0, t) = u_x(L, t) = 0, \quad (2)$$

y la condición inicial:

$$u(x, 0) = f(x) = 4x(L - x). \quad (3)$$

Se considera la representación truncada de Fourier:

$$u(x, t) = a_0 + \sum_{n=1}^N a_n e^{-k(2\pi n/L)^2 t} \cos\left(\frac{2n\pi x}{L}\right), \quad (4)$$

donde los coeficientes de Fourier a_n se calculan como:

$$a_n = \frac{2}{L} \int_0^L f(x) \cos\left(\frac{2n\pi x}{L}\right) dx. \quad (5)$$

Para la ecuación de onda:

$$u_{tt} = c^2 u_{xx}, \quad 0 < x < L, \quad t > 0.$$

con condiciones de frontera homogéneas de Neumann:

$$u_x(0, t) = 0, \quad u_x(L, t) = 0, \quad t > 0.$$

Condiciones iniciales:

$$\begin{aligned}u(x, 0) &= h(x), \quad 0 < x < L, \\u_t(x, 0) &= 0, \quad 0 < x < L.\end{aligned}$$

Usamos separación de variables y series de Fourier para obtener la solución:

$$u(x, t) = a_0 + \sum_{n=1}^{\infty} A_n \cos\left(\frac{2n\pi ct}{L}\right) \cos\left(\frac{2n\pi x}{L}\right),$$

con coeficientes:

$$A_n = \frac{2}{L} \int_0^L h(x) \cos\left(\frac{2n\pi x}{L}\right) dx.$$

Se implementan métodos computacionales en MATLAB/Octave para obtener las soluciones aproximadas y graficarlas.

2 Metodología

Se utilizarán las siguientes funciones implementadas en MATLAB/Octave:

2.1 Cálculo de los coeficientes de Fourier

Los coeficientes correspondientes a la representación aproximada de Fourier de las soluciones a las EDP previamente consideradas, pueden ser estimados utilizando el siguiente programa de propósito general.

```
function [a0,a,b] = F_g(f,N,L)
% Solucion aproximada de EDP del calor y de la onda por
% representacion truncada de Fourier
% Code by: Fredy Vides
    a0 = integral(f,0,L)/L;
    a = zeros(1,N);
    b = a;
    for n = 1:N
        fc = @(x) f(x).*cos(2*n*pi*x/L);
        fs = @(x) f(x).*sin(2*n*pi*x/L);
        a(n) = integral(fc,0,L)*2/L;
        b(n) = integral(fs,0,L)*2/L;
    end
end
```

2.2 Cálculo de las soluciones aproximadas

Las representación aproximada de Fourier de la solución a una EDP de calor como las consideradas previamente, puede estimarse utilizando el siguiente programa.

```

function u_hat = S_calor(x,t,k,a0,a,b,L)
% Solucion aproximada de EDP del calor por
% representacion truncada de Fourier
% Code by: Fredy Vides
    N = length(a);
    u_hat=a0;
    for n = 1:N
        u_hat = u_hat + ...
            (a(n)*cos(2*pi*n*x/L)+b(n)*sin(2*pi*n*x/L)).*exp(-k*((2*pi*n/L).^2*t));
    end

```

La representación aproximada de Fourier de la solución a una EDP de onda como las consideradas previamente, puede estimarse utilizando el siguiente programa.

```

function u_hat = S_onda(x,t,k,a0,a,b,c,d,L)
% Solucion aproximada de EDP del calor por
% representacion truncada de Fourier
% Code by: Fredy Vides
    N = length(a);
    u_hat=a0;
    for n = 1:N
        u_hat = u_hat + ...
            (a(n)*cos(2*pi*n*x/L)+b(n)*sin(2*pi*n*x/L)).*cos(2*k*pi*n*t/L)+...
            (c(n)*cos(2*pi*n*x/L)+d(n)*sin(2*pi*n*x/L)).*sin(2*k*pi*n*t/L);
    end

```

3 Ejemplo: Solución de la ecuación del calor

Se resuelve la ecuación:

$$u_t - k u_{xx} = 0, \quad 0 < x < L, \quad t > 0 \quad (6)$$

para $L = 1$, con condiciones de frontera homogéneas de Neumann:

$$u_x(0, t) = u_x(L, t) = 0, \quad (7)$$

y la condición inicial:

$$u(x, 0) = f(x) = 4x(L - x). \quad (8)$$

Se utilizan los siguientes comandos en MATLAB/Octave para resolver y graficar la solución aproximada:

```

>> [x,t]=meshgrid (0:1/100:1,0:2/100:.2);
>> f = @(x) 4*x.*(1-x);
>> [a0,a,b]=F_g(f,10,1);

```

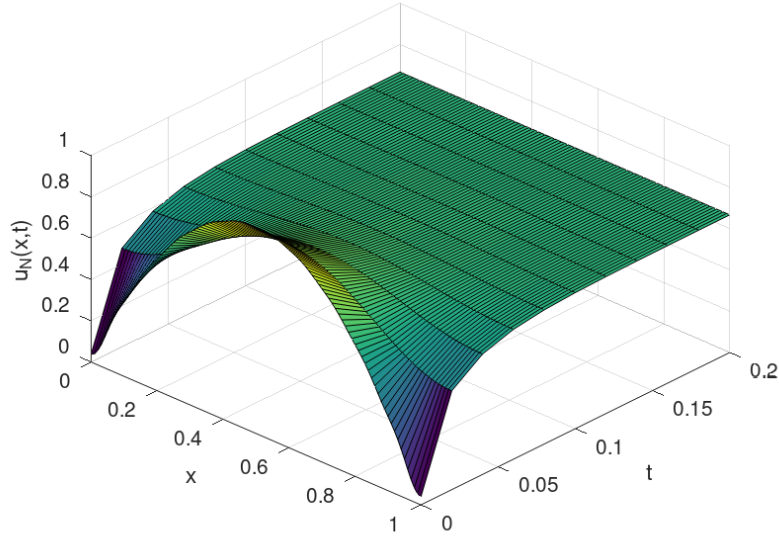


Figure 1: Representación gráfica de la solución aproximada de Fourier $u_N(x, t)$.

```
>> u_hat = S_calor(x,t,1,a0,a,b,1);
>> surf(x,t,u_hat)
>> xlabel ('x')
>> ylabel ('t')
>> zlabel ('u_N(x,t)')
```

La solución aproximada $u_N(x, t)$ se visualiza en la Figura 1.

4 Ejemplo: Solución de la ecuación de la onda

Se resuelve la ecuación:

$$u_{tt} - k^2 u_{xx} = 0, \quad 0 < x < L, \quad t > 0 \quad (9)$$

para $L = 1$, con condiciones de frontera homogéneas de Neumann:

$$u_x(0, t) = u_x(L, t) = 0, \quad (10)$$

condición inicial:

$$u(x, 0) = h(x) = 1 - |2x - 1|. \quad (11)$$

y la velocidad inicial:

$$u_t(x, 0) = 0. \quad (12)$$

Se utilizan los siguientes comandos en MATLAB/Octave para resolver y graficar la solución aproximada:

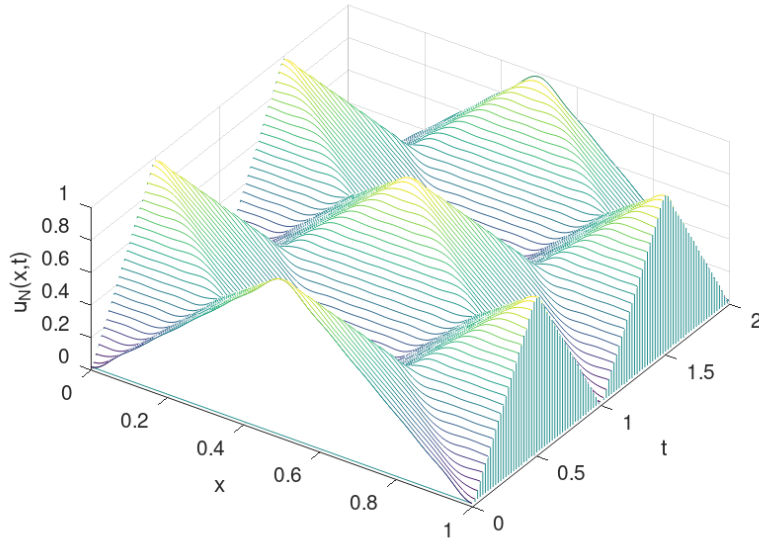


Figure 2: Representación gráfica de la solución aproximada de Fourier $u_N(x, t)$.

```
>> [x,t]=meshgrid (0:1/100:1,0:2/100:2);
>> h = @(x) 1-abs(2*x-1);
>> [a0,a,b]=F_g(h,10,1);
>> u_hat = S_onda(x,t,1,a0,a,b,0*a,0*b,1);
>> waterfall (x,t,u_hat)
>> xlabel ('x')
>> ylabel ('t')
>> zlabel ('u_N(x,t)')
```

La solución aproximada $u_N(x, t)$ se visualiza en la Figura 2.

5 Análisis de Resultados

- A medida que aumenta N , la representación truncada de Fourier mejora la precisión de la solución.

6 Conclusiones

Se implementó un método basado en series de Fourier para resolver la ecuación del calor y de la onda en una dimensión con condiciones de Neumann. Los resultados muestran que las soluciones preservan la simetría esperada y cumplen con la interpretación física de los problemas.