# Ecuaciones diferenciales parciales de segundo orden

Manuel Carlevaro

Departamento de Ingeniería Mecánica

Grupo de Materiales Granulares - UTN FRLP

manuel.carlevaro@gmail.com

Cálculo Avanzado • 2023

XeLaTeX

**Ecuación diferencial parcial (EDP) lineal de segundo orden:**

$$A\frac{\partial^2\phi}{\partial x^2} + B\frac{\partial^2\phi}{\partial x\partial y} + C\frac{\partial^2\phi}{\partial y^2} + D\frac{\partial\phi}{\partial x} + E\frac{\partial\phi}{\partial y} + F\phi = S$$

donde $A, B, C, D, E, F$ y $S$ son funciones de $x$ y $y$ en $D \in \mathbf{R}^2$.

**Ecuación diferencial parcial (EDP) lineal de segundo orden:**

$$A\frac{\partial^2\phi}{\partial x^2} + B\frac{\partial^2\phi}{\partial x\partial y} + C\frac{\partial^2\phi}{\partial y^2} + D\frac{\partial\phi}{\partial x} + E\frac{\partial\phi}{\partial y} + F\phi = S$$

donde $A, B, C, D, E, F$ y $S$ son funciones de $x$ y $y$ en $D \in \mathbf{R}^2$.

**Tipos:**

- Parabólica: $B^2 - 4AC = 0$
- Elíptica: $B^2 - 4AC < 0$
- Hiperbólica: $B^2 - 4AC > 0$

para todo $(x, y) \in D$.

**Casos:**

- Conducción de calor en sólidos, flujo de fluidos
- Ejemplos:
  - Conducción de calor:

$$\rho c\frac{\partial T}{\partial t} = k\frac{\partial^2 T(x,t)}{\partial x^2} + Q(x)$$

  - Transporte convectivo:

$$\frac{\partial\phi}{\partial t} = -\frac{\partial}{\partial x}u(x)\phi + D\frac{\partial^2\phi}{\partial x^2}$$

**Ecuación diferencial parcial (EDP) lineal de segundo orden:**

$$A\frac{\partial^2\phi}{\partial x^2} + B\frac{\partial^2\phi}{\partial x\partial y} + C\frac{\partial^2\phi}{\partial y^2} + D\frac{\partial\phi}{\partial x} + E\frac{\partial\phi}{\partial y} + F\phi = S$$

donde $A, B, C, D, E, F$ y $S$ son funciones de $x$ y $y$ en $D \in \mathbf{R}^2$.

**Tipos:**

- Parabólica: $B^2 - 4AC = 0$
- Elíptica: $B^2 - 4AC < 0$
- Hiperbólica: $B^2 - 4AC > 0$

para todo $(x, y) \in D$.

**Casos:**

- Problemas estacionarios de 2 y 3 dimensiones
- Conducción de calor en sólidos, vibración de membranas
- Ejemplos:
  - Ecuación de Poisson:

    $$-\nabla^2\phi(x, y) = S(x, y)$$

  - Ecuación de Laplace:

    $$-\nabla^2\phi(x, y) = 0$$

**Ecuación diferencial parcial (EDP) lineal de segundo orden:**

$$A\frac{\partial^2\phi}{\partial x^2} + B\frac{\partial^2\phi}{\partial x\partial y} + C\frac{\partial^2\phi}{\partial y^2} + D\frac{\partial\phi}{\partial x} + E\frac{\partial\phi}{\partial y} + F\phi = S$$

donde $A, B, C, D, E, F$ y $S$ son funciones de $x$ y $y$ en $D \in \mathbf{R}^2$.

**Tipos:**

- Parabólica: $B^2 - 4AC = 0$
- Elíptica: $B^2 - 4AC < 0$
- Hiperbólica: $B^2 - 4AC > 0$

para todo $(x, y) \in D$.

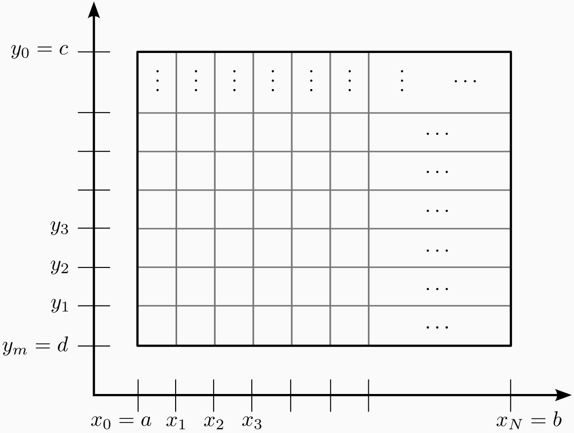
**Casos:**

- Problemas oscilatorios, propagación de ondas, fluidos
- Ejemplos:
  - Ecuación de onda:
    $$\frac{\partial^2 u(x,y,z,t)}{\partial t^2} = c^2\nabla^2 u(x,y,z,t)$$
  - Navier-Stokes (incompresible):
    $$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u}\cdot\nabla)\boldsymbol{u} - \nu\nabla^2\boldsymbol{u} = -\nabla\left(\frac{p}{p_0}\right) + \boldsymbol{g}$$

**Ecuación de Poisson (elíptica):**

$$\nabla^2 u(x,y) = \frac{\partial^2 u}{\partial x^2}(x,y) + \frac{\partial^2 u}{\partial y^2}(x,y) = f(x,y)$$

en $R = \{(x,y) | a < x < b, c < y < d\}$, con $u(x,y) = g(x,y)$ para $(x,y) \in S$, siendo $S$ la frontera de $R$.




**Malla:**

- División $[a,b]$ y $[c,d]$ en $n$ y $m$ partes iguales
- $h = (b-a)/n$, $k = (d-c)/m$
- $x_i = a + ih, \ i = 0, 1, \ldots, n$
- $y_j = c + jk, \ j = 0, 1, \ldots, m$

**Aproximación en diferencias finitas (serie de Taylor):**

$$\frac{\partial^2 u}{\partial x^2} = \frac{u(x_{i+1}, y_j) - 2u(x_i, y_j) + u(x_{i-1}, y_j)}{h^2} + \mathcal{O}(h^2)$$

$$\frac{\partial^2 u}{\partial y^2} = \frac{u(x_i, y_{j+1}) - 2u(x_i, y_j) + u(x_i, y_{j-1})}{k^2} + \mathcal{O}(k^2)$$

Con $u(x_i, y_j) \mapsto u_{i,j}$:

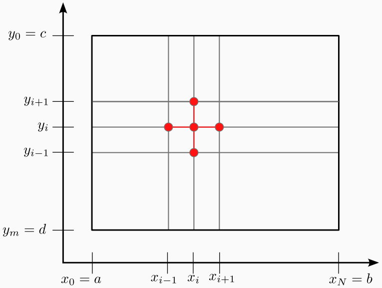
$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2} = f_{i,j} + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, y_j) + \frac{k^2}{12}\frac{\partial^4 u}{\partial y^4}(x_i, \eta_j)$$

para $i = 1, 2, \ldots, n-1$, $j = 1, 2, \ldots, m-1$ y condiciones de contorno:

$$u_{0,j} = g_{0,j} \quad \text{y} \quad u_{n,j} = g_{n,j}, \quad j = 0, 1, \ldots m;$$
$$u_{i,0} = g_{i,0} \quad \text{y} \quad u_{i,m} = g_{i,m}, \quad i = 0, 1, \ldots n$$

Resulta:

$$2\left[\left(\frac{h}{k}\right)^2 + 1\right] u_{i,j} - (u_{i+1,j} + u_{i-1,j}) - \left(\frac{h}{k}\right)^2 (u_{i,j+1} + u_{i,j-1}) = -h^2 f_{i,j},\ i \in [1, n-1],\ j \in [1, m-1]$$



Con $u(x_i, y_j) \mapsto u_{i,j}$:

$$\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2} = f_{i,j} + \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i, y_j) + \frac{k^2}{12}\frac{\partial^4 u}{\partial y^4}(x_i, \eta_j)$$

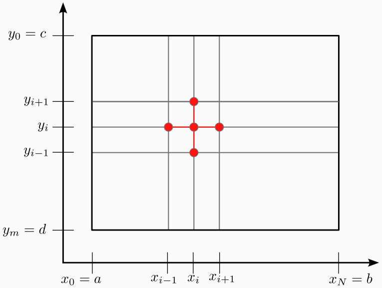
para $i = 1, 2, \ldots, n-1$, $j = 1, 2, \ldots, m-1$ y condiciones de contorno:

$$u_{0,j} = g_{0,j} \quad \text{y} \quad u_{n,j} = g_{n,j}, \quad j = 0, 1, \ldots m;$$

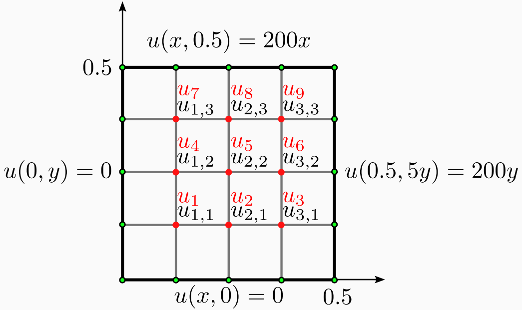
$$u_{i,0} = g_{i,0} \quad \text{y} \quad u_{i,m} = g_{i,m}, \quad i = 0, 1, \ldots n$$

Resulta:

$$2\left[\left(\frac{h}{k}\right)^2 + 1\right] u_{i,j} - (u_{i+1,j} + u_{i-1,j}) - \left(\frac{h}{k}\right)^2 (u_{i,j+1} + u_{i,j-1}) = -h^2 f_{i,j},\ i \in [1, n-1],\ j \in [1, m-1]$$




**Ejemplo:** Determinar la distribución estacionaria de temperaturas en una placa de $0.5 \times 0.5$ m usando $n = m = 4$. Dos bordes adyacentes se mantienen a 0 ℃ y la temperatura se incrementa linealmente en los otros bordes hasta llegar a 100 ℃ en la esquina de unión.




$$\frac{\partial^2 u}{\partial x^2}(x, y) + \frac{\partial^2 u}{\partial y^2}(x, y) = 0; \ (x, y) \in [0, 0.5]^2$$

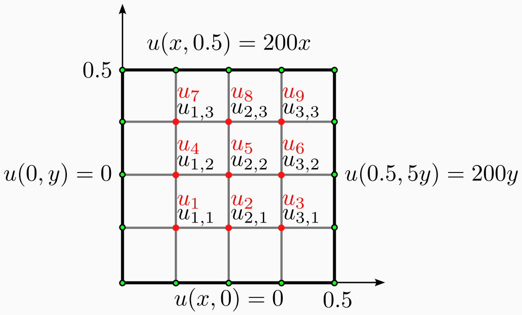
$$h = k = 1/8$$

$$u_{i,j} \mapsto u_l, \ l = i + m(j-1)$$

$$u_{1,1} = u_1, \ u_{2,1} = u_2, \ u_{3,1} = u_3$$

$$u_{1,2} = u_4, \ u_{2,2} = u_5, \ u_{3,2} = u_6$$

$$u_{1,3} = u_7, \ u_{2,3} = u_8, \ u_{3,3} = u_9$$




$$\frac{\partial^2 u}{\partial x^2}(x,y)+\frac{\partial^2 u}{\partial y^2}(x,y)=0;\ (x,y)\in[0,0.5]^2$$

$$h=k=1/8$$

Ecuaciones:

$$4u_{i,j}-u_{i+1,j}-u_{i-1,j}-u_{i,j-1}-u_{i,j+1}=0$$

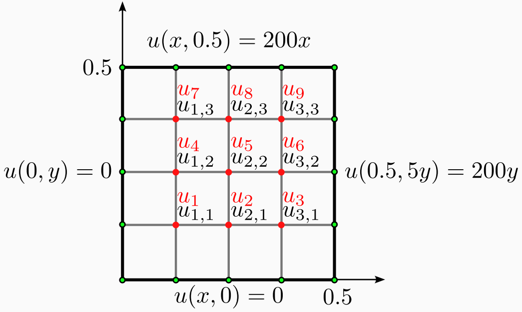
para $i=1,2,3;\ j=1,2,3.$

$$u_{i,j}\mapsto u_l,\ l=i+m(j-1)$$

$$u_{1,1}=u_1,\ u_{2,1}=u_2,\ u_{3,1}=u_3$$

$$u_{1,2}=u_4,\ u_{2,2}=u_5,\ u_{3,2}=u_6$$

$$u_{1,3}=u_7,\ u_{2,3}=u_8,\ u_{3,3}=u_9$$




$$\frac{\partial^2 u}{\partial x^2}(x,y)+\frac{\partial^2 u}{\partial y^2}(x,y)=0;\ (x,y)\in[0,0.5]^2$$

$$h=k=1/8$$

Ecuaciones:

$$4u_{i,j}-u_{i+1,j}-u_{i-1,j}-u_{i,j-1}-u_{i,j+1}=0$$

para $i=1,2,3;\ j=1,2,3$.

Condiciones de borde:

$$u_{0,0}=u_{0,1}=u_{0,2}=u_{0,3}=u_{0,4}=0$$

$$u_{1,0}=u_{2,0}=u_{3,0}=u_{4,0}=0$$

$$u_{1,4}=u_{4,1}=25; u_{2,4}=u_{4,2}=50$$

$$u_{3,4}=u_{4,3}=75; u_{4,4}=100$$

$$u_{i,j}\mapsto u_l,\ l=i+m(j-1)$$

$$u_{1,1}=u_1,\ u_{2,1}=u_2,\ u_{3,1}=u_3$$

$$u_{1,2}=u_4,\ u_{2,2}=u_5,\ u_{3,2}=u_6$$

$$u_{1,3}=u_7,\ u_{2,3}=u_8,\ u_{3,3}=u_9$$

$$
\begin{bmatrix}
4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{bmatrix}
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix}
=
\begin{bmatrix} u_{0,1} + u_{1,0} \\ u_{0,2} \\ u_{0,3} + u_{4,1} \\ u_{0,2} \\ 0 \\ u_{4,2} \\ u_{0,3} + u_{1,4} \\ u_{2,4} \\ u_{3,4} + u_{3,4} \end{bmatrix}
\rightarrow
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix}
=
\begin{bmatrix} 6.25 \\ 12.5 \\ 18.75 \\ 12.5 \\ 25 \\ 37.5 \\ 18.75 \\ 37.5 \\ 56.25 \end{bmatrix}
$$

$$
\begin{bmatrix}
4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\
-1 & 4 & -1 & 0 & -1 & 0 & 0 & 0 & 0 \\
0 & -1 & 4 & 0 & 0 & -1 & 0 & 0 & 0 \\
-1 & 0 & 0 & 4 & -1 & 0 & -1 & 0 & 0 \\
0 & -1 & 0 & -1 & 4 & -1 & 0 & -1 & 0 \\
0 & 0 & -1 & 0 & -1 & 4 & 0 & 0 & -1 \\
0 & 0 & 0 & -1 & 0 & 0 & 4 & -1 & 0 \\
0 & 0 & 0 & 0 & -1 & 0 & -1 & 4 & -1 \\
0 & 0 & 0 & 0 & 0 & -1 & 0 & -1 & 4
\end{bmatrix}
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix}
=
\begin{bmatrix} u_{0,1} + u_{1,0} \\ u_{0,2} \\ u_{0,3} + u_{4,1} \\ u_{0,2} \\ 0 \\ u_{4,2} \\ u_{0,3} + u_{1,4} \\ u_{2,4} \\ u_{3,4} + u_{3,4} \end{bmatrix}
\rightarrow
\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{bmatrix}
=
\begin{bmatrix} 6.25 \\ 12.5 \\ 18.75 \\ 12.5 \\ 25 \\ 37.5 \\ 18.75 \\ 37.5 \\ 56.25 \end{bmatrix}
$$

Solución exacta: $u(x, y) = 400xy$ por lo que la aproximación en diferencias finitas no tiene error:

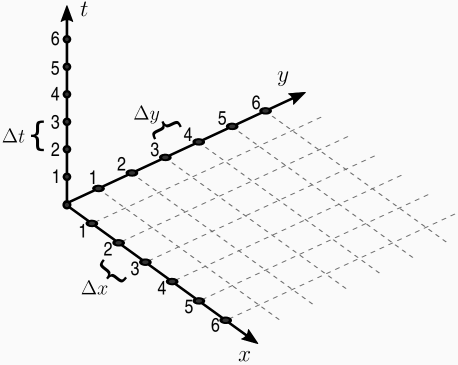
$$
\frac{\partial^4 u}{\partial x^4} = \frac{\partial^4 u}{\partial y^4} = 0
$$

**Ecuación de calor (parabólica):**

$$\frac{\partial u}{\partial t} = \alpha \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

**Ecuación de calor (parabólica):**

$$\frac{\partial u}{\partial t} = \alpha \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$

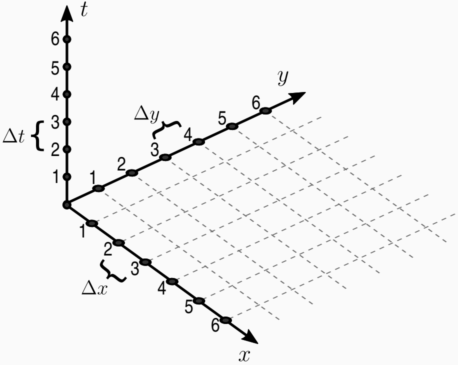



Grilla:

$$x_i = i\Delta x,\ y_j = j\Delta y,\ t_k = k\Delta t,\ u(x,y,t) = u_{i,j}^k$$

**Ecuación de calor (parabólica):**

$$\frac{\partial u}{\partial t} = \alpha \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right)$$




Grilla:

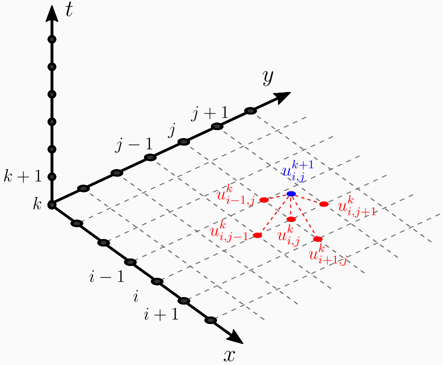
$$x_i = i\Delta x,\ y_j = j\Delta y,\ t_k = k\Delta t,\ u(x,y,t) = u_{i,j}^k$$

Diferencias finitas (hacia adelante - explícito):

$$\frac{u_{i,j}^{k+1} - u_{i,j}^k}{\Delta t} = \alpha \left( \frac{u_{i+1,j}^k - 2u_{i,j}^k + u_{i-1,j}^k}{\Delta x^2} \right. \\ \left. + \left( \frac{u_{i,j+1}^k - 2u_{i,j}^k + u_{i,j-1}^k}{\Delta y^2} \right) \right)$$

## Ecuación de calor (parabólica):

$$\frac{\partial u}{\partial t} = \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right)$$




Grilla:

$$x_i = i\Delta x,\ y_j = j\Delta y,\ t_k = k\Delta t,\ u(x,y,t) = u_{i,j}^k$$

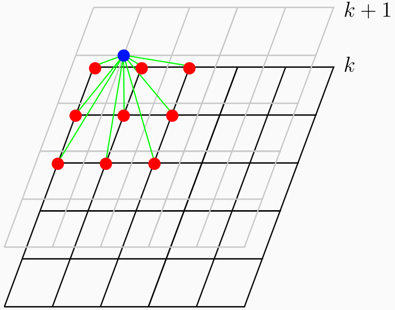
Diferencias finitas (hacia adelante - explícito):

$$\frac{u_{i,j}^{k+1} - u_{i,j}^k}{\Delta t} = \alpha\left(\frac{u_{i+1,j}^k - 2u_{i,j}^k + u_{i-1,j}^k}{\Delta x^2}\right.$$
$$\left. + \frac{u_{i,j+1}^k - 2u_{i,j}^k + u_{i,j-1}^k}{\Delta y^2}\right)$$

Hacemos $\Delta x = \Delta y$, $\gamma = \alpha\dfrac{\Delta t}{\Delta x^2}$:

$$u_{i,j}^{k+1} = \gamma\left(u_{i+1,j}^k + u_{i-1,j}^k + u_{i,j+1}^k + u_{i,j-1}^k - 4u_{i,j}^k\right) + u_{i,j}^k$$

Método explícito: $\Delta t \leq \dfrac{\Delta x^2}{4\alpha}$ $\longleftarrow$ estabilidad numérica.

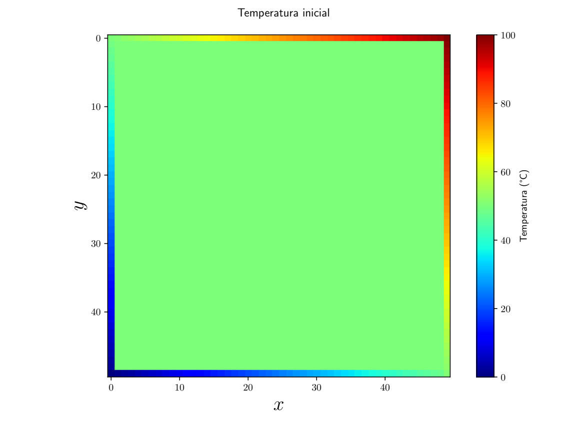



Stencil:

$$u_{i,j}^{k+1} = \gamma \left( u_{i+1,j}^{k} + u_{i-1,j}^{k} + u_{i,j+1}^{k} + u_{i,j-1}^{k} - 4u_{i,j}^{k} \right) + u_{i,j}^{k}$$

$$\frac{\partial u}{\partial t} - \alpha\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\right) = 0$$

$$0 \leq x \leq L_x,\ 0 \leq y \leq L_y$$




Condiciones de borde:

$$u(x,0) = 50 + \frac{x(100-50)}{L_x}$$

$$u(0,y) = 50 - y\frac{50}{L_y}$$

$$u(x,L_y) = 50\frac{x}{L_x}$$

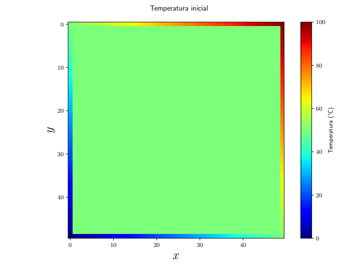
$$u(L_x,y) = 100 - y\frac{50}{L_y}$$

Condición inicial:

$$u(x,y)_{t=0} = 50$$

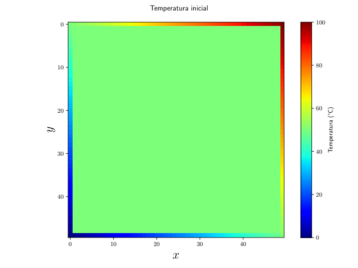
```python
#!/usr/bin/env python3

import numpy as np

Lx, Ly = 50, 50
max_iter_tiempo = 750
alpha = 5
delta_x = 1
delta_t = (delta_x ** 2)/(4 * alpha)
gamma = (alpha * delta_t) / (delta_x ** 2)

# Condiciones de borde
u_SO = 0.0
u_NO, u_SE = 50.0, 50.0
u_NE = 100.0

# Condición inicial interior
u_inicial = 50
```

```python
def inicializar_u(max_iter_tiempo, ni=Lx, nj=Ly):
    # Inicializar la solución: u(k, i, j)
    u = np.full((max_iter_tiempo, ni, nj), u_inicial)
    # Establecer condiciones de borde
    u[:, 0, :] = u_NO + np.arange(ni) * delta_x
                * (u_NE - u_NO) / Lx
    u[:, :, 0] = u_NO - np.arange(nj) * delta_x
                * u_NO / Ly
    u[:, -1, :] = np.arange(ni) * delta_x * u_SE / Lx
    u[:, :, -1] = u_NE - np.arange(nj) * delta_x
                * u_SE / Ly
    return u

u = inicializar_u(max_iter_tiempo)
```

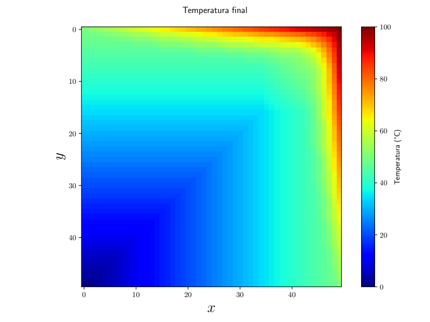


```python
import matplotlib
import matplotlib.pyplot as plt
matplotlib.rcParams.update({"text.usetex": True})
fig, ax = plt.subplots(figsize=(8,6))
mappable = ax.imshow(u[0], interpolation=None,
                     cmap=plt.cm.jet)
fig.colorbar(mappable, label="Temperatura (°C)", ax=ax)
ax.set_xlabel(r"$x$", fontsize=20)
ax.set_ylabel(r"$y$", fontsize=20)
fig.suptitle("Temperatura inicial")
fig.tight_layout()
fig.savefig("temp-inicial.pdf")
plt.close()
```

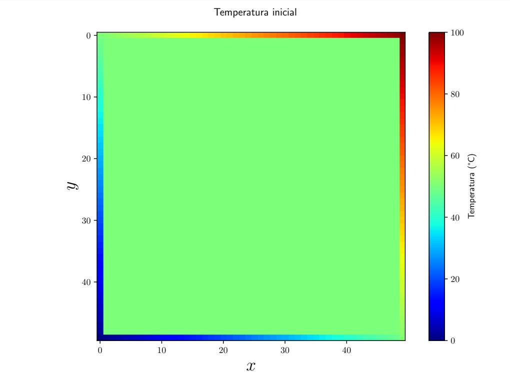
```python
# Código que aplica el stencil en la grilla (i, j) y en cada t
def calcular(u):
    nk, ni, nj = u.shape
    for k in range(0, nk-1):
        for i in range(1, ni-1):
            for j in range(1, nj-1):
                u[k + 1, i, j] = gamma * (u[k][i+1][j] + u[k][
                                          + u[k][i][j+1] + u[k
                                          - 4*u[k][i][j]) + u[

    return u

u = calcular(u)
```



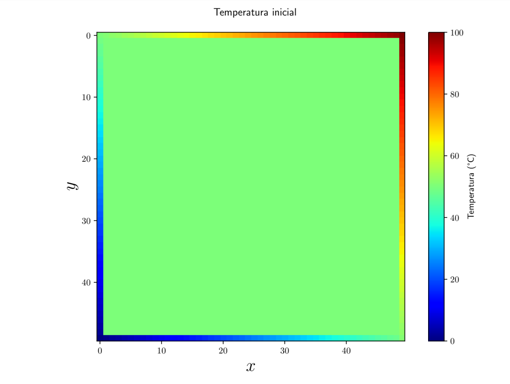
```python
# Código que aplica el stencil en la grilla (i, j) y en cada tiempo k
def calcular(u):
    nk, ni, nj = u.shape
    for k in range(0, nk-1):
        for i in range(1, ni-1):
            for j in range(1, nj-1):
                u[k + 1, i, j] = gamma * (u[k][i+1][j] + u[k][i-1][j]
                                          + u[k][i][j+1] + u[k][i][j-1]
                                          - 4*u[k][i][j]) + u[k][i][j]
    return u

u = calcular(u)
fig, ax = plt.subplots(figsize=(8,6))
mappable = ax.imshow(u[-1], interpolation=None,
                     cmap=plt.cm.jet)
fig.colorbar(mappable, label="Temperatura (°C)", ax=ax)
ax.set_xlabel(r"$x$", fontsize=20)
ax.set_ylabel(r"$y$", fontsize=20)
fig.suptitle("Temperatura final")
fig.tight_layout()
fig.savefig("temp-final.pdf")
plt.close()
```



```python
def plotheatmap(u_k, k):
    # Limpiamos la figura
    plt.clf()

    plt.title(f"Temperatura en t = {k*delta_t:.2f} u.t.")
    plt.xlabel(r"$x$", fontsize=20)
    plt.ylabel(r"$y$", fontsize=20)

    # Ploteamos u_k (u_{i,j} en `paso de tiempo k)
    plt.imshow(u_k, cmap=plt.cm.jet,
               interpolation="bicubic", vmin=0, vmax=100)
    plt.colorbar()

    return plt

import matplotlib.animation as animation
from matplotlib.animation import FuncAnimation

def animate(k):
    plotheatmap(u[k], k)

anim = animation.FuncAnimation(plt.figure(),
    animate, interval=50, frames=max_iter_tiempo,
    repeat=False)
anim.save("solucion_ecuacion_calor_t.mp4")
```



```python
def plotheatmap(u_k, k):
    # Limpiamos la figura
    plt.clf()

    plt.title(f"Temperatura en t = {k*delta_t:.2f} u.t.")
    plt.xlabel(r"$x$", fontsize=20)
    plt.ylabel(r"$y$", fontsize=20)

    # Ploteamos u_k (u_{i,j} en `paso de tiempo k)
    plt.imshow(u_k, cmap=plt.cm.jet,
               interpolation="bicubic", vmin=0, vmax=100)
    plt.colorbar()

    return plt

import matplotlib.animation as animation
from matplotlib.animation import FuncAnimation

def animate(k):
    plotheatmap(u[k], k)

anim = animation.FuncAnimation(plt.figure(),
    animate, interval=50, frames=max_iter_tiempo,
    repeat=False)
anim.save("solucion_ecuacion_calor_t.mp4")
```



**Análisis de estabilidad:**

$$\frac{\partial u}{\partial t}(x,t)=\alpha\frac{\partial^2 u}{\partial x^2}(x,t),\ 0<x<l,\ 0<t$$

Condiciones de frontera:

$$u(0,t)=u(l,t)=0,t>0;$$
$$u(x,0)=f(x),0\leq x\leq l$$

Diferencia hacia adelante (o progresiva):

$$\frac{\partial u}{\partial t}(x_i,t_j)=\frac{u(x_i,t_j+k)-u(x_i,t_j)}{k}-\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_j)$$
$$\frac{\partial^2 u}{\partial x^2}(x_i,t_j)=\frac{u(x_i+h,t_j)-2u(x_i,t_j)+u(x_i-h,t_j)}{h^2}$$
$$-\frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i,t_j)$$

con $t_j\in(t_j,t_{j+1}),\xi_i\in(x_i,x_{i+1})$

Resulta:

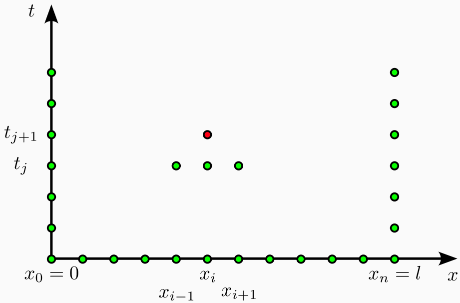
$$\frac{u_{i,j+1}-u_{i,j}}{k}-\alpha\frac{u_{i+1,j}-2u_{i,j}+u_{i-1,j}}{h^2}=0$$

con un error local de truncación:

$$\epsilon_{i,j}=\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_j)-\alpha^2\frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i,t_j)$$

Solución explícita:

$$u_{i,j+1}=\left(1-\frac{2\alpha k}{h^2}\right)u_{i,j}+\alpha\frac{k}{h^2}(u_{i+1,j}+u_{i-1,j})$$



Matriz $(n-1)\times(n-1)$:

$$\boldsymbol{A} = \begin{bmatrix} (1-2\lambda) & \lambda & 0 & \cdots & 0 \\ \lambda & (1-2\lambda) & \lambda & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \lambda \\ 0 & \cdots & 0 & \lambda & (1-2\lambda) \end{bmatrix}$$

con $\lambda = \alpha k/h^2$

Si $\boldsymbol{u}^{(\mathbf{0})} = (f(x_1), f(x_2), \cdots, f(x_{n-1}))^T$, la solución aproximada es:

$$\boldsymbol{u}^{(j)} = \boldsymbol{A}\boldsymbol{u}^{(j-1)}$$

Supongamos un error $\boldsymbol{e}^{(0)} = (e_1^{(0)}, e_2^{(0)}, \cdots, e_{n-1}^{(0)})^T$:

$$\boldsymbol{u}^{(1)} = \boldsymbol{A}(\boldsymbol{u}^{(0)} + \boldsymbol{e}^{(0)}) = \boldsymbol{A}\boldsymbol{u}^{(0)} + \boldsymbol{A}\boldsymbol{e}^{(0)}$$

Para el paso $k$, el error en $\boldsymbol{u}^{(k)} = \boldsymbol{A}^k\boldsymbol{e}^{(0)}$. El método es estable si $\|\boldsymbol{A}^k\boldsymbol{e}^{(0)}\| \leq \|\boldsymbol{e}^{(0)}\|$

$$\|\boldsymbol{A}^k\| \leq 1 \implies \rho(\boldsymbol{A}^k) = (\rho(\boldsymbol{A}))^k \leq 1$$

Autovalores de $\boldsymbol{A}$:

$$\mu_i = 1 - 4\lambda\left(\operatorname{sen}\left(\frac{i\pi}{2n}\right)\right)^2$$

Norma $L_\infty$:

$$\rho(\boldsymbol{A}) = \max_{1\leq i\leq n}\left|1 - 4\lambda\left(\operatorname{sen}\left(\frac{i\pi}{2n}\right)\right)^2\right|$$

que se simplifica a

$$0 \leq \lambda\left(\operatorname{sen}\left(\frac{i\pi}{2n}\right)\right)^2 \leq \frac{1}{2},\ i = 1, 2, \ldots, n-1$$

Esta desigualdad debe valer cuando $h \to 0, n \to \infty$:

$$\lim_{n\to\infty}\left[\operatorname{sen}\left(\frac{(n-1)\pi}{2n}\right)\right)^2\right] = 1$$

Por lo tanto habrá estabilidad si $0 \leq \lambda \leq 1/2$:

$$\alpha\frac{k}{h^2} \leq \frac{1}{2} \qquad \leftarrow \text{condicionalmente estable.}$$

**Ejemplo:**

$$\frac{\partial u}{\partial t}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t),\ 0 < x < l,\ 0 < t$$

$$u(0,t) = u(l,t) = 0, t > 0;$$

$$u(x,0) = \operatorname{sen}(\pi x), 0 \le x \le 1$$

- con $h = 0.1$ y $k = 0.0005$. (1000 pasos)
- con $h = 0.1$ y $k = 0.01$. (50 pasos)

Solución exacta:

$$u(x,t) = e^{-\pi^2 t} \operatorname{sen}(\pi x)$$

Solución: `parabolica-progresiva.py`

| $i$ | $x_i$ | $u_{i,50}$ | $u(x_i, 50)$ | $\lvert u_{i,50} - u(x_i, 50)\rvert$ |
|---|---|---|---|---|
| 0 | 0.0 | 0.00000 | 0.00000 | 0.000e+00 |
| 1 | 0.1 | 0.00229 | 0.00222 | 6.411e-05 |
| 2 | 0.2 | 0.00435 | 0.00423 | 1.219e-04 |
| 3 | 0.3 | 0.00599 | 0.00582 | 1.678e-04 |
| 4 | 0.4 | 0.00704 | 0.00684 | 1.973e-04 |
| 5 | 0.5 | 0.00740 | 0.00719 | 2.075e-04 |
| 6 | 0.6 | 0.00704 | 0.00684 | 1.973e-04 |
| 7 | 0.7 | 0.00599 | 0.00582 | 1.678e-04 |
| 8 | 0.8 | 0.00435 | 0.00423 | 1.219e-04 |
| 9 | 0.9 | 0.00229 | 0.00222 | 6.411e-05 |
| 10 | 1.0 | 0.00000 | 0.00000 | 8.808e-19 |

**Ejemplo:**

$$\frac{\partial u}{\partial t}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t),\ 0 < x < l,\ 0 < t$$

$$u(0,t) = u(l,t) = 0, t > 0;$$

$$u(x,0) = \operatorname{sen}(\pi x), 0 \leq x \leq 1$$

- con $h = 0.1$ y $k = 0.0005$. (1000 pasos)
- con $h = 0.1$ y $k = 0.01$. (50 pasos)

Solución exacta:

$$u(x,t) = e^{-\pi^2 t} \operatorname{sen}(\pi x)$$

Solución: `parabolica-progresiva.py`

| $i$ | $x_i$ | $u_{i,50}$ | $u(x_i, 50)$ | $\lvert u_{i,50} - u(x_i, 50)\rvert$ |
|---|---|---|---|---|
| 0 | 0.0 | 0.000e+00 | 0.00000 | 0.000e+00 |
| 1 | 0.1 | 2.637e+05 | 0.00222 | 2.637e+05 |
| 2 | 0.2 | -5.026e+05 | 0.00423 | 5.026e+05 |
| 3 | 0.3 | 6.938e+05 | 0.00582 | 6.938e+05 |
| 4 | 0.4 | -8.186e+05 | 0.00684 | 8.186e+05 |
| 5 | 0.5 | 8.643e+05 | 0.00719 | 8.643e+05 |
| 6 | 0.6 | -8.254e+05 | 0.00684 | 8.254e+05 |
| 7 | 0.7 | 7.047e+05 | 0.00582 | 7.047e+05 |
| 8 | 0.8 | -5.135e+05 | 0.00423 | 5.135e+05 |
| 9 | 0.9 | 2.704e+05 | 0.00222 | 2.704e+05 |
| 10 | 1.0 | 0.000e+00 | 0.00000 | 8.808e-19 |

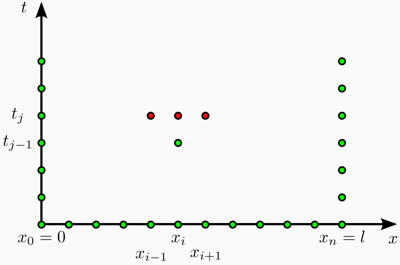
**Condicionalmente estable:** diferencias regresivas (implícito).

$$\frac{\partial u}{\partial t}(x_i, t_j) = \frac{u(x_i, t_j) - u(x_i, t_{j-1})}{k} - \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j)$$

con $\mu_j \in (t_{j-1}, t_j)$.
Reemplazando en la ecuación en derivadas parciales:

$$\frac{u_{i,j} - u_{i,j-1}}{k} - \alpha\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = 0$$

para $i = 1, 2, \ldots, n-1$ y $j = 1, 2, \ldots$.




Hacemos $\lambda = \alpha k/h^2$:

$$(1+2\lambda)u_{i,j} - \lambda u_{i+1,j} - \lambda u_{i-1,j} = u_{i,j-1}$$

Con las condiciones de frontera:

$$u_{i,0} = f(x_i),\ i = 1, 2, \ldots, n-1$$
$$u_{0,j} = u_{n,j} = 0,\ j = 1, 2, \ldots$$

Matriz $(n-1)\times(n-1)$:

$$\boldsymbol{A} = \begin{bmatrix} (1+2\lambda) & -\lambda & 0 & \cdots & 0 \\ -\lambda & (1+2\lambda) & -\lambda & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & -\lambda \\ 0 & \cdots & 0 & -\lambda & (1+2\lambda) \end{bmatrix}$$

$\boldsymbol{u}^{(j)} = (u_{1,j}, u_{2,j}, \cdots, u_{n-1,j})^T$,
$\boldsymbol{u}^{(j-1)} = (u_{1,j-1}, u_{2,j-1}, \cdots, u_{n-1,j-1})^T$

$$\mapsto \boldsymbol{A}\boldsymbol{u}^{(j)} = \boldsymbol{u}^{(j-1)},\ j = 1, 2, \ldots$$

**Ejemplo:**

$$\frac{\partial u}{\partial t}(x,t) = \alpha\frac{\partial^2 u}{\partial x^2}(x,t),\ 0 < x < 1,\ 0 < t$$

$$u(0,t) = u(1,t) = 0, t > 0;$$

$$u(x,0) = \operatorname{sen}(\pi x), 0 \leq x \leq 1$$

con $h = 0.1$ y $k = 0.01$.

Solución exacta:

$$u(x,t) = e^{-\pi^2 t}\operatorname{sen}(\pi x)$$

Solución: `parabolica-regresiva.py`

| $i$ | $x_i$ | $u_{i,50}$ | $u(x_i, 50)$ | $\lvert u_{i,50} - u(x_i, 50)\rvert$ |
|---|---|---|---|---|
| 0 | 0.0 | 0.00000 | 0.00000 | 0.000e+00 |
| 1 | 0.1 | 0.00780 | 0.00222 | 5.576e-03 |
| 2 | 0.2 | 0.01236 | 0.00423 | 8.133e-03 |
| 3 | 0.3 | 0.01591 | 0.00582 | 1.010e-02 |
| 4 | 0.4 | 0.01817 | 0.00684 | 1.133e-02 |
| 5 | 0.5 | 0.01894 | 0.00719 | 1.175e-02 |
| 6 | 0.6 | 0.01817 | 0.00684 | 1.133e-02 |
| 7 | 0.7 | 0.01591 | 0.00582 | 1.010e-02 |
| 8 | 0.8 | 0.01236 | 0.00423 | 8.133e-03 |
| 9 | 0.9 | 0.00780 | 0.00222 | 5.576e-03 |
| 10 | 1.0 | 0.00000 | 0.00000 | 8.808e-19 |

**Ejemplo:**

$$\frac{\partial u}{\partial t}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t),\ 0 < x < 1,\ 0 < t$$

$$u(0,t) = u(1,t) = 0, t > 0;$$

$$u(x,0) = \operatorname{sen}(\pi x), 0 \leq x \leq 1$$

con $h = 0.1$ y $k = 0.01$.

Solución exacta:

$$u(x,t) = e^{-\pi^2 t} \operatorname{sen}(\pi x)$$

Solución: `parabolica-regresiva.py`

Autovalores de $\boldsymbol{A}$:

$$\mu_i = 1 + 4\lambda \left[\operatorname{sen}\left(\frac{i\pi}{2n}\right)\right]^2$$

para $i = 1, 2, \cdots, n-1$. Como $\lambda > 0$, $\mu_i > 0$.

| $i$ | $x_i$ | $u_{i,50}$ | $u(x_i, 50)$ | $\lvert u_{i,50} - u(x_i, 50)\rvert$ |
|---|---|---|---|---|
| 0 | 0.0 | 0.00000 | 0.00000 | 0.000e+00 |
| 1 | 0.1 | 0.00780 | 0.00222 | 5.576e-03 |
| 2 | 0.2 | 0.01236 | 0.00423 | 8.133e-03 |
| 3 | 0.3 | 0.01591 | 0.00582 | 1.010e-02 |
| 4 | 0.4 | 0.01817 | 0.00684 | 1.133e-02 |
| 5 | 0.5 | 0.01894 | 0.00719 | 1.175e-02 |
| 6 | 0.6 | 0.01817 | 0.00684 | 1.133e-02 |
| 7 | 0.7 | 0.01591 | 0.00582 | 1.010e-02 |
| 8 | 0.8 | 0.01236 | 0.00423 | 8.133e-03 |
| 9 | 0.9 | 0.00780 | 0.00222 | 5.576e-03 |
| 10 | 1.0 | 0.00000 | 0.00000 | 8.808e-19 |

Entonces $\rho(\boldsymbol{A}^{-1}) < 1 \mapsto \boldsymbol{A}$ es una matriz convergente:

$$\lim_{j\to\infty} (\boldsymbol{A}^{-1})^j \boldsymbol{e}^{(0)} = \boldsymbol{0}$$ $\leftarrow$ incondicionalmente estable.

Precisión: $\mathcal{O}(k + h^2)$.

**Método de Crank-Nicolson**:

Diferencias progresivas:

$$\frac{u_{i,j+1} - u_{i,j}}{k} - \alpha \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = 0$$

con error de truncamiento:

$$\epsilon_p = \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) + \mathcal{O}(h^2)$$

Diferencias regresivas:

$$\frac{u_{i,j+1} - u_{i,j}}{k} - \alpha \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2} = 0$$

con error de truncamiento:

$$\epsilon_r = -\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \hat{\mu}_j) + \mathcal{O}(h^2)$$

Suponiendo que:

$$\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) \approx \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \hat{\mu}_j)$$

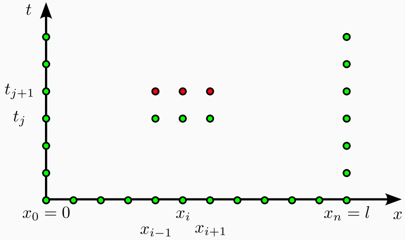
Suponiendo que:

$$\frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \mu_j) \approx \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i, \hat{\mu}_j)$$

el método de la diferencia promediado:

$$\frac{u_{i,j+1} - u_{i,j}}{k} - \frac{\alpha}{2}\left[\frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} + \frac{u_{i+1,j+1} - 2u_{i,j+1} + u_{i-1,j+1}}{h^2}\right] = 0$$

tiene un error de truncamiento $\mathcal{O}(k^2 + h)$.



En forma matricial:

$$\boldsymbol{A}\boldsymbol{u}^{(j+1)} = \boldsymbol{B}\boldsymbol{u}^{(j)}, \quad \text{para cada} \quad j = 0, 1, \ldots, \quad \text{donde} \quad \lambda = \alpha\frac{k}{h^2},\ \boldsymbol{u}^{(j)} = (u_{1,j}, u_{2,j}, \cdots, u_{n-1,j})^T$$

y

$$\boldsymbol{A} = \begin{bmatrix} (1+\lambda) & -\lambda/2 & 0 & \cdots & 0 \\ -\lambda/2 & (1+\lambda) & -\lambda/2 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & -\lambda/2 \\ 0 & \cdots & 0 & -\lambda/2 & (1+\lambda) \end{bmatrix} \quad \text{y} \quad \boldsymbol{B} = \begin{bmatrix} (1-\lambda) & \lambda/2 & 0 & \cdots & 0 \\ \lambda/2 & (1-\lambda) & \lambda/2 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \lambda/2 \\ 0 & \cdots & 0 & \lambda/2 & (1-\lambda) \end{bmatrix}$$

**Ejemplo:**

$$\frac{\partial u}{\partial t}(x,t) = \alpha \frac{\partial^2 u}{\partial x^2}(x,t),\ 0 < x < 1,\ 0 < t$$

$$u(0,t) = u(1,t) = 0, t > 0;$$

$$u(x,0) = \operatorname{sen}(\pi x), 0 \leq x \leq 1$$

con $h = 0.1$ y $k = 0.01$.

Solución exacta:

$$u(x,t) = e^{-\pi^2 t}\operatorname{sen}(\pi x)$$

Solución: `parabolica-regresiva.py`

| $i$ | $x_i$ | $u_{i,50}$ | $u(x_i, 50)$ | $\lvert u_{i,50} - u(x_i, 50)\rvert$ |
|---|---|---|---|---|
| 0 | 0.0 | 0.00000 | 0.00000 | 0.000e+00 |
| 1 | 0.1 | 0.00488 | 0.00222 | 2.660e-03 |
| 2 | 0.2 | 0.00812 | 0.00423 | 3.895e-03 |
| 3 | 0.3 | 0.01066 | 0.00582 | 4.842e-03 |
| 4 | 0.4 | 0.01228 | 0.00684 | 5.436e-03 |
| 5 | 0.5 | 0.01283 | 0.00719 | 5.639e-03 |
| 6 | 0.6 | 0.01228 | 0.00684 | 5.436e-03 |
| 7 | 0.7 | 0.01066 | 0.00582 | 4.842e-03 |
| 8 | 0.8 | 0.00812 | 0.00423 | 3.895e-03 |
| 9 | 0.9 | 0.00488 | 0.00222 | 2.660e-03 |
| 10 | 1.0 | 0.00000 | 0.00000 | 8.808e-19 |

## Ecuación de onda (hiperbólica)

$$\frac{\partial^2 u}{\partial t^2}(x,t) - \alpha^2 \frac{\partial^2 u}{\partial x^2}(x,t) = 0,\ 0 < x < l,\ 0 < t$$

Con las ecuaciones de frontera:

$$u(0,t) = u(l,t) = 0, t > 0;$$

$$u(x,0) = f(x),\ \frac{\partial u}{\partial t}(x,0) = g(x);\ 0 \le x \le l$$

Malla:

$$x_i = ih,\ t_j = jk$$

para $i = 0, 1, \ldots, n$ y $j = 0, 1, \cdots$

En todos los puntos de la malla interior:

$$\frac{\partial^2 u}{\partial t^2}(x_i,t_j) - \alpha^2 \frac{\partial^2 u}{\partial x^2}(x_i,t_j) = 0$$

Diferencias centrales para las derivadas segundas:

$$\frac{\partial^2 u}{\partial t^2}(x_i,t_j) = \frac{u(x_i,t_{j+1}) - 2u(x_i,t_j) + u(x_i,t_{j-1})}{k^2} - \frac{k^2}{12}\frac{\partial^4 u}{\partial t^4}(x_i,\mu_j)$$

$$\frac{\partial^2 u}{\partial x^2}(x_i,t_j) = \frac{u(x_{i+1},t_j) - 2u(x_i,t_j) + u(x_{i-1},t_j)}{h^2} - \frac{h^2}{12}\frac{\partial^4 u}{\partial x^4}(\xi_i,t_j)$$

donde $\xi_i \in (x_{i-1}, x_{i+1}), \mu_i \in (t_{j-1}, t_{j+1})$. Resulta:

$$\frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{k^2} - \alpha^2 \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{h^2} = \epsilon_{i,j}$$

con

$$\epsilon_{i,j} = \frac{1}{12}\left[k^2 \frac{\partial^4 u}{\partial t^4}(x_i,t_j) - \alpha^2 h^2 \frac{\partial^4 u}{\partial x^4}(x_i,t_j)\right]$$

Ignorando $\epsilon_{i,j}$ y haciendo $\lambda = \alpha k/h$ (número de Courant):

$$u_{i,j+1} - 2u_{i,j} + u_{i,j-1} - \lambda^2 u_{i+1,j} + 2\lambda^2 u_{i,j} - \lambda^2 u_{i-1,j} = 0$$

Resolviendo para el paso temporal:

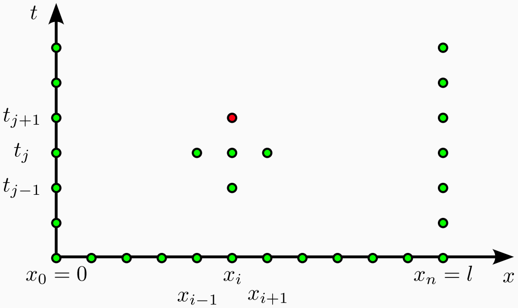
$$u_{i,j+1} = 2(1-\lambda^2)u_{i,j} + \lambda^2(u_{i+1,j} + u_{i-1,j}) - u_{i,j-1}$$

para $i = 1, 2, \ldots, n-1$ y $j = 1, 2, \ldots$.
Las condiciones de frontera resultan:

$$u_{0,j} = u_{n,j} = 0,\ j = 1, 2, \ldots$$

y la condición inicial:

$$u_{i,0} = f(x_i),\ i = 1, 2, \ldots, n-1$$




Enfoque matricial:

$$\boldsymbol{u}^{(j+1)} = \boldsymbol{A}\boldsymbol{u}^{(j)} - \boldsymbol{u}^{(j-1)}$$

$$\begin{bmatrix} u_{1,j+1} \\ u_{2,j+1} \\ u_{3,j+1} \\ \vdots \\ u_{n-1,j+1} \end{bmatrix} = \begin{bmatrix} 2(1-\lambda^2) & \lambda^2 & 0 & \cdots & 0 \\ \lambda^2 & 2(1-\lambda^2) & \lambda^2 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \lambda^2 \\ 0 & \cdots & 0 & \lambda^2 & 2(1+\lambda^2) \end{bmatrix} \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ u_{3,j} \\ \vdots \\ u_{n-1,j} \end{bmatrix} - \begin{bmatrix} u_{1,j-1} \\ u_{2,j-1} \\ u_{3,j-1} \\ \vdots \\ u_{n-1,j-1} \end{bmatrix}$$

Problema: ¿$u_{i,1}$?

Condición de velocidad inicial:

$$\frac{\partial u}{\partial t}(x,0) = g(x), \ 0 \leq x \leq l$$

Aproximación por diferencias progresivas:

$$\frac{\partial u}{\partial t}(x,0) = \frac{u(x_i,t_1) - u(x_i,0)}{k} - \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_i)$$

Resolviendo:

$$u(x_i,t_1) = u(x_i,0) + kg(x_i) + \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_i)$$

Error de truncamiento: $\mathcal{O}(k)$

Mejora: expansión de McLaurin ($\sim$ Taylor):

$$u(x_i,t_1) = u(x_i,0) + k\frac{\partial u}{\partial t}(x_i,0) + \frac{k^2}{2}\frac{\partial^2 u}{\partial t^2}(x_i,0) + \frac{k^3}{6}\frac{\partial^3 u}{\partial t^3}(x_i,\hat{\mu}_i)$$

Si $f''$ existe:

$$\frac{\partial^2 u}{\partial t^2}(x_i,0) = \alpha^2\frac{\partial^2 u}{\partial x^2}(x_i,0) = \alpha^2 f''(x_i)$$

entonces:

$$u(x_i,t_1) = u(x_i,0) + kg(x_i) + \frac{\alpha^2 k^2}{2}f''(x_i)$$

con error $\mathcal{O}(k^3)$.

$$\begin{bmatrix} u_{1,j+1} \\ u_{2,j+1} \\ u_{3,j+1} \\ \vdots \\ u_{n-1,j+1} \end{bmatrix} = \begin{bmatrix} 2(1-\lambda^2) & \lambda^2 & 0 & \cdots & 0 \\ \lambda^2 & 2(1-\lambda^2) & \lambda^2 & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \cdots & \cdots & \cdots & \lambda^2 \\ 0 & \cdots & 0 & \lambda^2 & 2(1+\lambda^2) \end{bmatrix} \begin{bmatrix} u_{1,j} \\ u_{2,j} \\ u_{3,j} \\ \vdots \\ u_{n-1,j} \end{bmatrix} - \begin{bmatrix} u_{1,j-1} \\ u_{2,j-1} \\ u_{3,j-1} \\ \vdots \\ u_{n-1,j-1} \end{bmatrix}$$

Problema: ¿$u_{i,1}$?

Condición de velocidad inicial:

$$\frac{\partial u}{\partial t}(x,0) = g(x),\ 0 \leq x \leq l$$

Aproximación por diferencias progresivas:

$$\frac{\partial u}{\partial t}(x,0) = \frac{u(x_i,t_1) - u(x_i,0)}{k} - \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_i)$$

Resolviendo:

$$u(x_i,t_1) = u(x_i,0) + kg(x_i) + \frac{k}{2}\frac{\partial^2 u}{\partial t^2}(x_i,\mu_i)$$

Error de truncamiento: $\mathcal{O}(k)$

Mejora: expansión de McLaurin ($\sim$ Taylor):

$$u(x_i,t_1) = u(x_i,0) + k\frac{\partial u}{\partial t}(x_i,0) + \frac{k^2}{2}\frac{\partial^2 u}{\partial t^2}(x_i,0) + \frac{k^3}{6}\frac{\partial^3 u}{\partial t^3}(x_i,\hat{\mu}_i)$$

Si $f''$ existe:

$$\frac{\partial^2 u}{\partial t^2}(x_i,0) = \alpha^2\frac{\partial^2 u}{\partial x^2}(x_i,0) = \alpha^2 f''(x_i)$$

entonces:

$$u(x_i,t_1) = u(x_i,0) + kg(x_i) + \frac{\alpha^2 k^2}{2}f''(x_i)$$

con error $\mathcal{O}(k^3)$. ¿Y si no tenemos $f''(x_i)$?

Ecuación en diferencias:

$$f''(x_i) = \frac{f(x_{i+1}) - 2f(x_i) + f(x_{i-1})}{h^2} - \frac{h^2}{12} f^{(4)}(\xi_i)$$

Usando $\lambda = \alpha k/h$:

$$u(x_i, 1) = (1 - \lambda^2) f(x_i) + \frac{\lambda^2}{2} f(x_{i+1}) + \frac{\lambda^2}{2} f(x_{i-1}) + kg(x_i) + \mathcal{O}(k^3 + h^2 k^2)$$

Entonces, para $i = 1, 2, \ldots, n-1$ usamos:

$$u_{i,1} = (1 - \lambda^2) f(x_i) + \frac{\lambda^2}{2} f(x_{i+1}) + \frac{\lambda^2}{2} f(x_{i-1}) + kg(x_i)$$

y la ecuación matricial para $j = 2, 3, \ldots$

- R.L. Burden, D.J. Faires y A.M. Burden. ***Análisis numérico.*** 10.ª ed. Mexico: Cengage Learning, 2017. Capítulo 12.
- E. Kreyszig, H. Kreyszig y E.J. Norminton. ***Advanced Engineering Mathematics.*** Hoboken, USA: John Wiley & Sons, Inc, 2011. Capítulo 21.
- H.P. Langtangen y S. Linge. ***Finite Difference Computing with PDEs - A Modern Software Approach.*** https://hplgit.github.io/fdm-book/doc/web/index.html. 2016. Capítulo 3.