

Sesión 1: Técnicas Computacionales Avanzadas de Hidro/Aero-Dinámica en Ingeniería y Astrofísica

M.Sc. Erick Urquilla - eurquilla@vols.utk.edu

Doctorando en Física, Departamento de Física y Astronomía,
Universidad de Tennessee, Knoxville, EE. UU.

July 23, 2025

1 Github Codespaces

Utilizaremos Github codespaces para compilar y ejecutar las simulaciones en EMU.

1. Ingresa al repositorio de EMU en el siguiente enlace: <https://github.com/erickurquilla1999/inspirastem-2025-cfd>.
2. Haz clic en el botón verde **Code** y, en la pestaña *Codespaces*, crea un Codespace en la rama **wsues**.
3. Una vez creado el Codespace, deberás ver una sesión de Visual Studio Code en tu navegador. Codespace corre en el sistema operativo *Ubuntu 22.04 LTS* (64 bit) alojado en la nube de GitHub.

2 Comandos básicos de Ubuntu

La terminal que ves en Visual Studio Code dentro de GitHubCodespaces es una terminal Ubuntu. Aquí tienes algunos comandos que utilizarás durante este workshop; prácticalos hasta que te familiarices con ellos.

- ls -lah** Lista el contenido del directorio actual (*l*) en formato largo, con tamaños legibles (*h*) e incluye archivos ocultos (*a*).
- pwd** Muestra la ruta completa (*print working directory*) del directorio en el que te encuentras.
- cd** Cambia de directorio. Ej. **cd /Documentos** navega a la carpeta **Documentos** en tu *home*.

`mkdir` Crea un directorio nuevo. Ej. `mkdir proyecto`.

`cp` Copia archivos o carpetas. Ej. `cp archivo.txt copia.txt`. Usa `-r` para copiar directorios.

`mv` Mueve o renombra archivos/directorios. Ej. `mv viejo.txt nuevo.txt`.

`rm -r` Elimina archivos o carpetas de forma recursiva. ****Irreversible;**** úsalo con cuidado.

3 Requisitos para ejecutar el código

Antes de ejecutar el código, debemos instalar algunos paquetes que utiliza el programa. Ejecuta los siguientes comandos:

- **Instalar librerías de Python**
`pip install imageio`

4 Generación de malla

Para resolver este ejercicio, ve al cuaderno llamado `sesion1.ipynb` en la carpeta `sesion1` del repositorio.

La generación de una malla es fundamental en el método de elementos finitos. La malla divide el dominio espacial en elementos finitos más pequeños, lo que permite aproximar la solución de la ecuación diferencial en cada uno de estos elementos. En el método *Galerkin Discontinuo*, cada elemento puede tener su propia solución polinómica, lo que permite capturar discontinuidades y variaciones abruptas en las variables del fluido.

Ejercicio 1

Dadas las coordenadas inicial `x_inicial`, final `x_final`, el número de elementos `N_elementos` y el número de nodos por elemento `N_nodos`, codifica una malla cartesiana unidimensional con las coordenadas de los nodos interiores de cada elemento (elementos y nodos igualmente espaciados en una malla regular). El numpy array resultante debe llamarse `mall`. La forma del array `mall` debe ser $(N_elementos, N_nodos)$. Es decir, el componente `mall[i, j]` representa la coordenada en x del nodo j del elemento i .

Como ejemplo, observa la siguiente imagen de una malla con `N_elementos = 3` y `N_nodos = 4`. El arreglo `mall` contiene las posiciones de los nodos interiores de cada elemento.

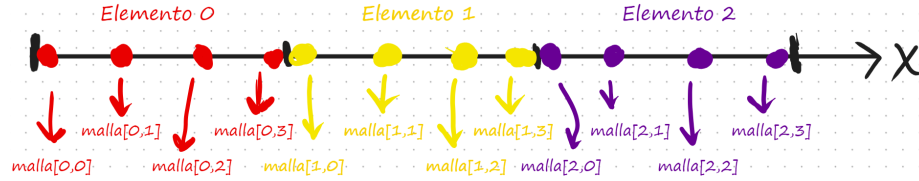


Figure 1: Ejemplo de malla para `N_elementos = 3` y `N_nodos = 4`.

4.1 Condiciones iniciales

Para resolver este ejercicio, ve al cuaderno llamado `sesion1.ipynb` en la carpeta `sesion1` del repositorio.

Para practicar, las condiciones iniciales del fluido las estableceremos como

$$h_0(x) = 1 + 0.1 e^{-(x-5)^2}, \quad u_0(x) = 0.$$

Las fronteras del dominio del fluido se encuentran en $x = 0$ m y $x = 10$ m.

Ejercicio 2

Crea dos arreglos de NumPy que contengan las condiciones iniciales:

1. Un arreglo llamado `h` de forma $(N_elementos, N_nodos)$. El elemento `h[i, j]` representará la altura (en metros) del fluido en el nodo j del elemento i .
2. Un arreglo llamado `u` de forma $(N_elementos, N_nodos)$. El elemento `u[i, j]` representará la velocidad (en metros por segundo) del fluido en el nodo j del elemento i .