Proyecto Final

Juan David Argüello Plata

Universidad Industrial de Santander
Escuela de Ingeniería Mecánica

El presente trabajo trata sobre el estudio temporal-dinámico que sufre un río cuando sortea un obstáculo; como una piedra, por ejemplo.

Figura 1. Río obstaculizado.

1. Geometría

La geometría de estudio se puede apreciar en la Figura 2.

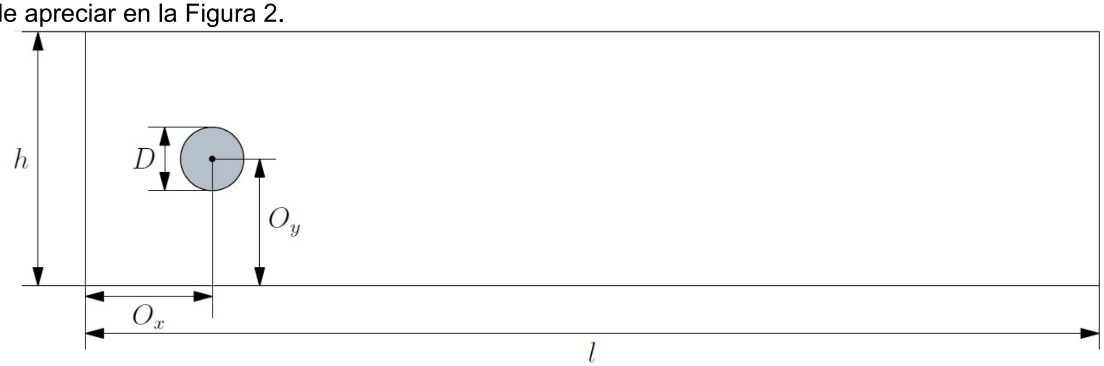


Figura 2. Geometría.

2. Ecuaciones de Navier-Stokes

Las ecuaciones de Navier-Stokes para flujo incompresible formado por las ecuaciones de velocidad u y presión p se pueden apreciar en la Ecuación 1.

$$\rho\left(\frac{\partial u}{\partial t}+u\cdot\nabla u\right)=\nabla\cdot\sigma(u,p)+f, \tag{1}$$
$$\nabla\cdot u=0$$

De la ecuación de momento (Ecuación 1), f se refiere a la fuerza dada por unidad de volumen, $\sigma(u,p)$ se refiere al tensor de esfuerzos, que para un fluido newtoniano está dado por:

$$\sigma(u,p)=2\mu\epsilon(u)-pI \tag{2}$$

Siendo ϵ el tensor de deformaciones.

$$\epsilon(u)=\frac{1}{2}\left(\nabla u+(\nabla u)^T\right) \tag{3}$$

μ es la viscosidad dinámica.

2.1. Forma Variacional

Para evitar solucionar el sistema de ecuaciones mediante precondicionadores y metodologías iterativas, se empleará un acercamiento matemático conocido como el *método de división*. Existen diferentes estrategias de división para flujo incompresible de las ecuaciones de Navier-Stokes, uno de ellos es el *método de Chorin* (IPCS).

Involucra tres etapas. En primer lugar, se propone una velocidad tentativa u^* al emplear la ecuación de momento (1) como una diferencia finita centrada en el tiempo, pero usando la presión p^n del tiempo anterior. Además, se linealiza el el término convectivo no lineal al emplear la velocidad conocida u^n del tiempo anterior $(u^n\cdot\nabla u^n)$. $$ \begin{equation} \rho \left<\left(u^* - u^n \right) / \Delta t, v \right> + \left<\rho u^n \cdot \nabla u^n, v \right> + \left< \sigma \left(u^{n+ \frac{1}{2}}, p^n \right), \epsilon (v) \right> \\ + \left _{\partial \Omega} - \left<\mu \nabla u ^{n+\frac{1}{2} \cdot n, v} \right> _{\partial \Omega} = \left \tag{4} \label{Chorin} \end{equation}

$$DelaEcuación???, lanotación\$ \langle v,w\rangle \$setratadelosiguiente:$$

$$\langle v,w\rangle=\int_\Omega vw\,dx \tag{5}$$
$$\langle v,w\rangle_{\partial\Omega}=\int_{\partial\Omega} vw\,ds$$

$$u^{n+\frac{1}{2}}\approx\frac{u^n+u^{n+1}}{2} \tag{6}$$

$$ Se puede observar que el *problema variacional* (Ecuación ???) surge de la integración por partes al término $\langle-\nabla\cdot\sigma,v\rangle$; dónde:

$$\langle-\nabla\cdot\sigma,v\rangle=\langle\sigma,\epsilon(v)\rangle-\langle T,v\rangle_{\partial\Omega} \tag{7}$$

Siendo $T=\sigma\cdot n$ la tracción límite. Si se resuelve el problema con una frontera libre, se puede tomar $T=0$. Sin embargo, si se calcula el flujo a través de un canal o tubería y se desea modelar un "canal imaginario" en la salida, se necesita tratar este término con sumo cuidado. Para este caso, se realizaría la suposición de que la derivada de la velocidad en la dirección del canal es cero a la salida (condición de frontera tipo *Neumann*), que corresponde a un flujo completamente desarrollado o que no cambia de manera significativa.

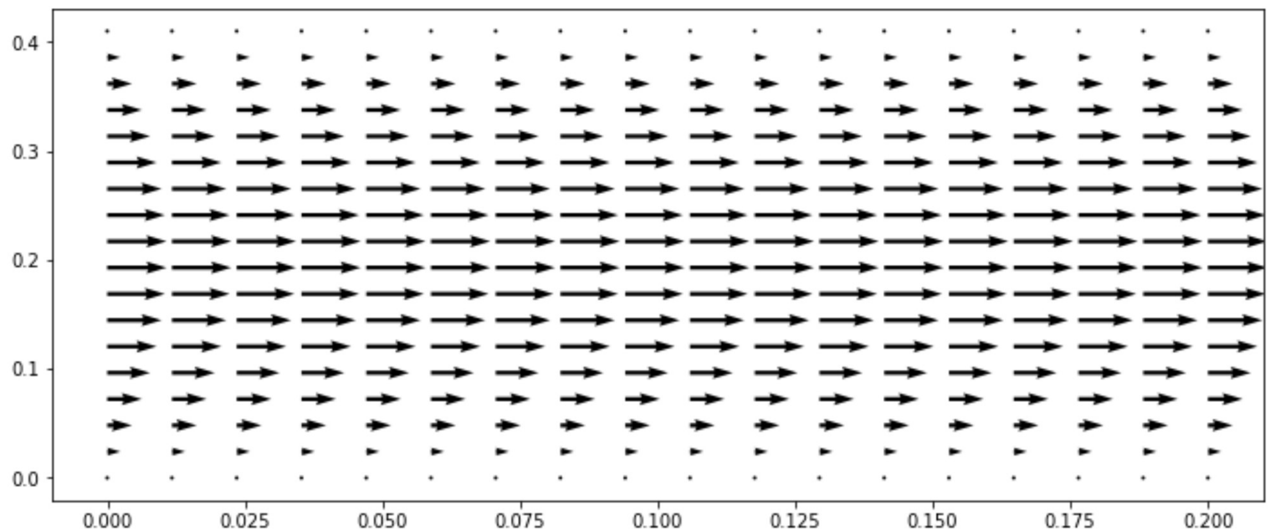
3. Perfil de velocidades

El perfil de velocidades inicial del río se supone como conocido y presenta el comportamiento descrito en la Ecuacón 11.

$$u(x,y,t) = \left(1.5 \cdot \frac{Cy(h-y)}{h^2}, 0\right) \tag{11}$$

In [2]:

```
%matplotlib inline
from App.Read import Read
from App.Velocidades import Perfil
#Datos
datos = Read(data)
#Perfil de velocidades
Perfil(datos)
```



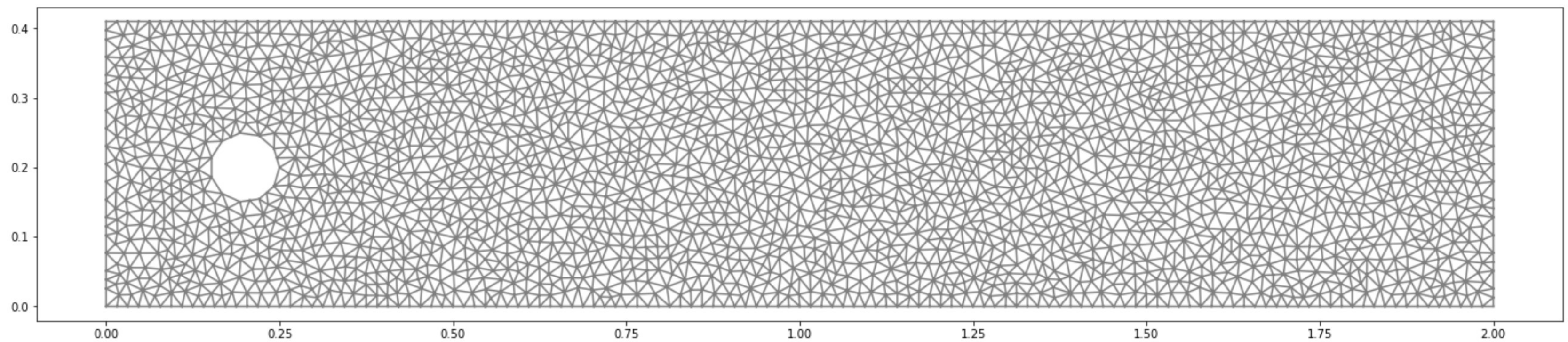
4. Malla

La malla de la geometría se crea ejecutando el siguiente código:

In [18]:
```
%matplotlib inline
from IPython.display import display
from ipywidgets import *
from fenics import *
from mshr import *
import numpy as np
import matplotlib.pyplot as plt

datos = Read(data)
# Geometría y malla
fig = plt.figure(figsize=(40,5))
canal = Rectangle(Point(0,0), Point(datos['Geo']['l [m]'], datos['Geo']['h [m]']))
piedra = Circle(Point(datos['Geo']['Ox [m]'], datos['Geo']['Oy [m]']), datos['Geo']['D [m]']/2)
rio = canal - piedra
mesh = generate_mesh(rio, 64)
mesha['malla'] = mesh
plot(mesh)
```

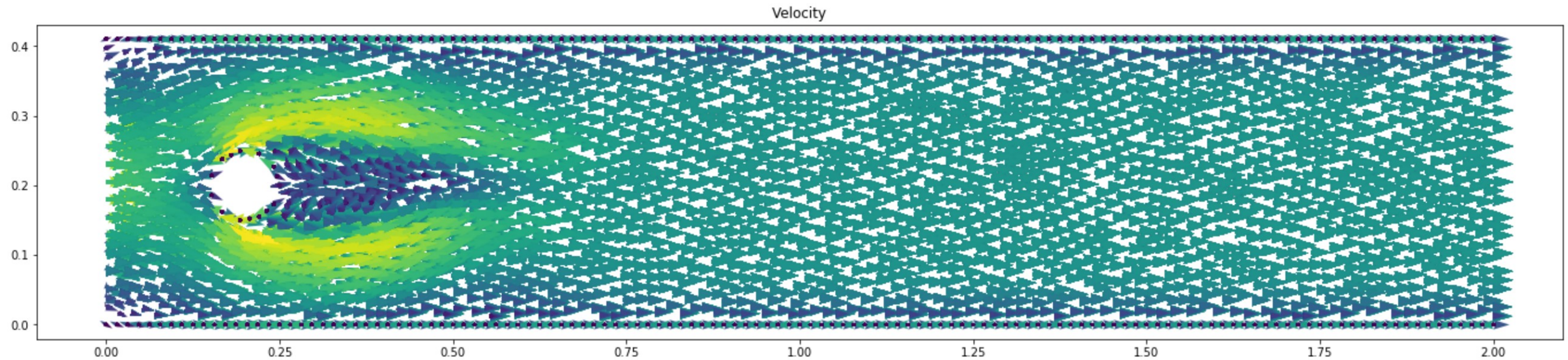
Out[18]:
```
[<matplotlib.lines.Line2D at 0x7fc5b01fadd8>,
 <matplotlib.lines.Line2D at 0x7fc5b01faf28>]
```



5. Solución

La solución del problema de interés, con la malla especificada, se puede apreciar a continuación.

In [15]:
```
%matplotlib inline
from App.Sol import *
datos = Read(data)
Solution(datos, mesha['malla'], size=(40,5))
```



6. Simulación dinámica

A continuación, se puede apreciar una simulación dinámica del problema de interés.

In [16]:
```
%gui tk
from App.Sim import *
Simulation(datos)
```