# MECH 570C-FSI: Code Project 1
# Fluid-structure interaction of a rigid circular cylinder

Jincong Li
60539939

Feb 26th

## Abstract

This project explores the fundamental aspects of fluid-structure interaction (FSI) with a specific focus on vortex-induced vibrations (VIV) emanating from the flow around a rigid circular cylinder. Utilizing the incompressible, two-dimensional Navier-Stokes equations within the Arbitrary Lagrangian-Eulerian (ALE) framework, the study aims to accurately model the interaction between fluid dynamics and the rigid body structure endowed with two degrees of translational freedom. Two primary objectives are pursued: the implementation of an IntegratedOutput function for calculating forces on the cylinder and the development of an ALE mesh to represent mesh movement accurately. The methodology encompasses the application of Gauss Quadrature, Galerkin Projection methods, and the formulation of the ALE mesh displacement equations. The results from the IntegratedOutput function demonstrate high fidelity with reference values, confirming the accuracy of the force calculation method. However, the ALE mesh implementation exhibited discrepancies, particularly in simulating VIV scenarios, indicating the necessity for further refinement. The project's findings contribute to the understanding of fluid-structure dynamics and underscore the challenges and complexities inherent in accurately modeling FSI phenomena.

# Introduction

This project is designed to develop foundational knowledge in the field of fluid-structure interaction, focusing on a paradigmatic issue of vortex-induced vibration caused by flow around a smooth circular cylinder. The fluid dynamics are characterized using the incompressible, two-dimensional Navier-Stokes equations, formulated within the arbitrary Lagrangian-Eulerian (ALE) framework. The analysis of the rigid body structure, which possesses a two-degree-of-freedom translational capability, is conducted employing the Lagrangian description. At the interface between the fluid and structure, both velocity continuity and traction equilibrium conditions are rigorously maintained.

# Methodology

Given that a substantial portion of the code has been provided, two primary objectives have been delineated as follows: The initial goal involves the execution of the IntegratedOutput function by completing the requisite equations to calculate the tractions and forces exerted on the cylinder's surface. The secondary task is to formulate the code for the Arbitrary Lagrangian-Eulerian (ALE) mesh, thereby enabling the accurate representation of the mesh's movement. Detailed instructions for the implementation of these two tasks will be elaborated upon in subsequent subsections. Moreover, the overall logic of the entire code is discussed in the last subsection.

## Implementing IntegratedOutput

The IntegratedOutput function initiates the process by identifing the indices of elements located on the cylinder's surface, subsequently extracting the global coordinates of each node from the matrix containing global coordinate information for subsequent application. Following this, the Gauss Quadrature and Galerkin Projection methods, in conjunction with the shape function, are applied to discretize the governing equation of the stress tensor along the boundary of each element on the cylinder, represented as:

$$\sigma = -p\mathbf{I} + \mu(\nabla\mathbf{U} + \nabla\mathbf{U}^T),$$

whereafter the integrals are computed for each element. The traction in x-direction ($\mathbf{t_x}$) is the first element in $\sigma$ and the traction in y-direction ($\mathbf{t_y}$) is the second elements. Thus the code-wise view is:

$$\begin{aligned} tx(:,p) &= ((2 * fluid.visc. * locgradUx - locP). * normal(:,1) \\ &\quad + fluid.visc. * (locgradUy + locgradVx). * normal(:,2)); \\ ty(:,p) &= ((2 * fluid.visc. * locgradVy - locP). * normal(:,2) \\ &\quad + fluid.visc. * (locgradUy + locgradVx). * normal(:,1)); \end{aligned}$$

The next step is to multiply those tractions for each quadrature point as well as for each element on the cylinder boundary with their corresponding surface area and then report the forces as the output of the IntegratedOutput function.

The verification of this objective is discussed in the Result section.

## Implementing ALEmesh

The alemesh function starts from initializing the ALE displacement of each nodes in the global domain and forming the local to global map. Boundary values of ALE displacement are identified to be zero on the outter boundaries and setted equal to the corresponding displacement on the cylinder boundary. Then with the similar implementing strategy as shown in navierstoke function, Galerkinterms function, and Poisson example, the governing equation of ALE mehs displacement:

$$\begin{aligned} \nabla \cdot \sigma^{\mathbf{m}} = 0, \in \Omega^s \\ \sigma = \nabla \eta + \nabla \eta^T + (\nabla \cdot \eta)\mathbf{I} \end{aligned}$$

is firstly transformed into weak form:

$$\int_{\Omega} \big(\nabla v \cdot \nabla \eta + \nabla v \cdot (\nabla \eta)^T + (\nabla \cdot \eta)(\nabla \cdot v)\big)\, d\Omega = 0$$

Then by applying the Gauss Quadrature and Galerkin Projection methods with the shape function, the corresponding terms are represented by stiffness matrix (please refer to the code for detail information) and the displacement for free nodes are solved and returned as the output of alemesh function. Note the velocity of ALE mesh movement is also computed since the navierstoke function requires it.

## Overall Logic

The overall logic of the code is explained as follows: In each timestep in main function, the IntegratedOutput function computes the force on the cylinder surface at current location, and then the rigidbody function convert the forces into displacement of the cylinder boundary. ALEmesh function then computed new displacement for all nodes other than the outter boundaries and cylinder boundary according to the displacement of the cylinder boundary and compute the velocity of mesh movement. Thus, the global coordinates of each nodes are updated in the main function. Finally, the navierstoke function takes the updated coordinates information and mesh velocity as input, it returns corresponding fluid velocity and pressure distribution.

In the next timestep, all information from the last timestep is stored as "previous" version and repeat the procedure indicated above.

# Results

## Verification of IntegratedOutput

By setting Reynolds number to 100 (viscosity of the fluid is set to be 5 instead of 10) and disable the rigidbody function and alemesh function (equivalent to a stationary cylinder), the code runs for 2000 timesteps with $\Delta t = 0.1s$. The following is the coefficient analysis and their plots. Since the simulated

Table 1: IntegratedOutput Result Comparison

| | Reference Values | Simulated Values |
|---|---|---|
| $\bar{C}_d$ | 1.375 | 1.3753 |
| $C_l^{\max}$ | 0.3352 | 0.3359 |
| $\bar{C}_l^{rms}$ | 0.2368 | 0.2375 |

values agree with the reference values, one can conclude the implementation of the IntegratedOutput function is correct. Also the plots of coefficient of lift and drag are shown below in Figure 1 & 2.

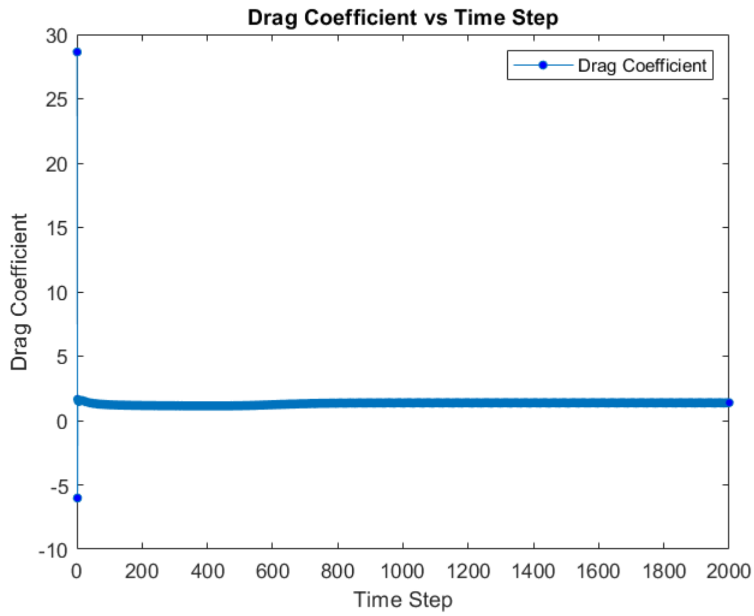


Figure 1: Drag Coefficient

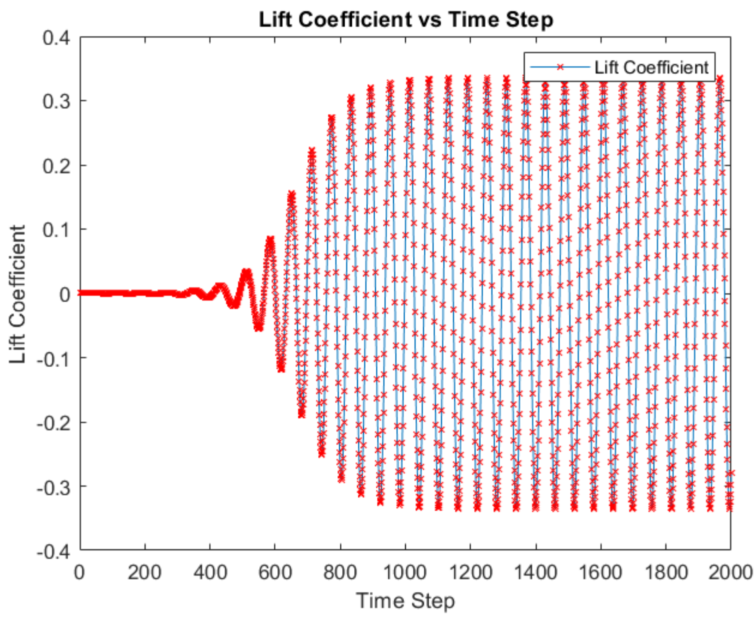


Figure 2: Lift Coefficient

## Verification of ALE mesh

The alemesh function is tested by executing a simulation for a two-degree-of-freedom vortex-induced vibration (VIV) scenario of the cylinder, devoid of any damping coefficient, at a Reynolds number ($Re$) of 200, featuring a mass ratio $m^* = \frac{4m}{\rho f D^2} = 10$; and a reduced velocity $U_r = \frac{U}{f_n D} = 5$, where $m$ denotes the mass of the cylinder, and $f_n = \frac{\sqrt{k/m}}{2\pi}$ represents the natural frequency of the elastically mounted cylinder with $k = k_x = k_y$ being the spring stiffness. The outcomes of this simulation can be compared with the results shown in the Table 2. Note that the simulation of the VIV problem is not quite

Table 2: VIV Problem Result Comparison

| | Reference Values[1] | Simulated Values |
|---|---|---|
| $\bar{C}_d$ | 2.0551 | 1.1369 |
| $\bar{C}_l^{rms}$ | 0.0893 | 0.5062 |

correct. In the simulation, the cylinder tends to flutter after 50 seconds, so the simulated value shown in Table 2 is computed from the first 500 steps. The overall coefficients plots are shown in Figure 3 & 4 below. Mesh and contour

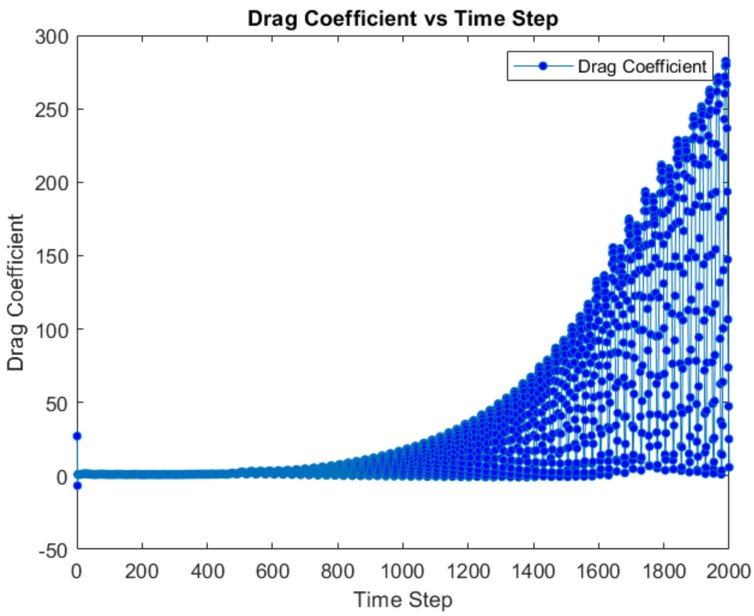


Figure 3: Drag Coefficient

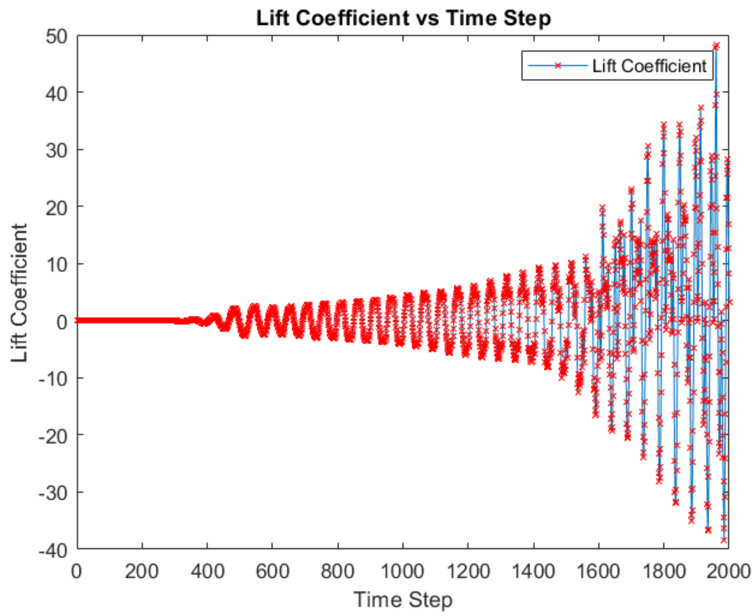


Figure 4: Lift Coefficient

representations at various instances are delineated from Figures 5 to 8, within which one can discern the cylinder undergoing flutter. Consequently, the computation of coefficients is rendered inaccurate and becomes nonsensical. Given that the IntegratedOutput function has been validated for correctness, the issue likely resides within an improper implementation within the alemesh function. The structural and logical framework of the alemesh function bears resemblance to both the Galerkinterms function and the Poisson example, suggesting a potential oversight in accurately incorporating the Galerkin term associated with the third term of the weak form in the ALE governing equation. Additionally, the integration of boundary conditions into the alemesh function may not be as straightforward as initially coded and could necessitate adjustments. While other errors may exist beyond my current recognition, the extensive duration required to execute the code for diagnostic insights—coupled with the substantial time and effort required for a line-by-line examination—precluded the resolution of all errors by the conclusion.

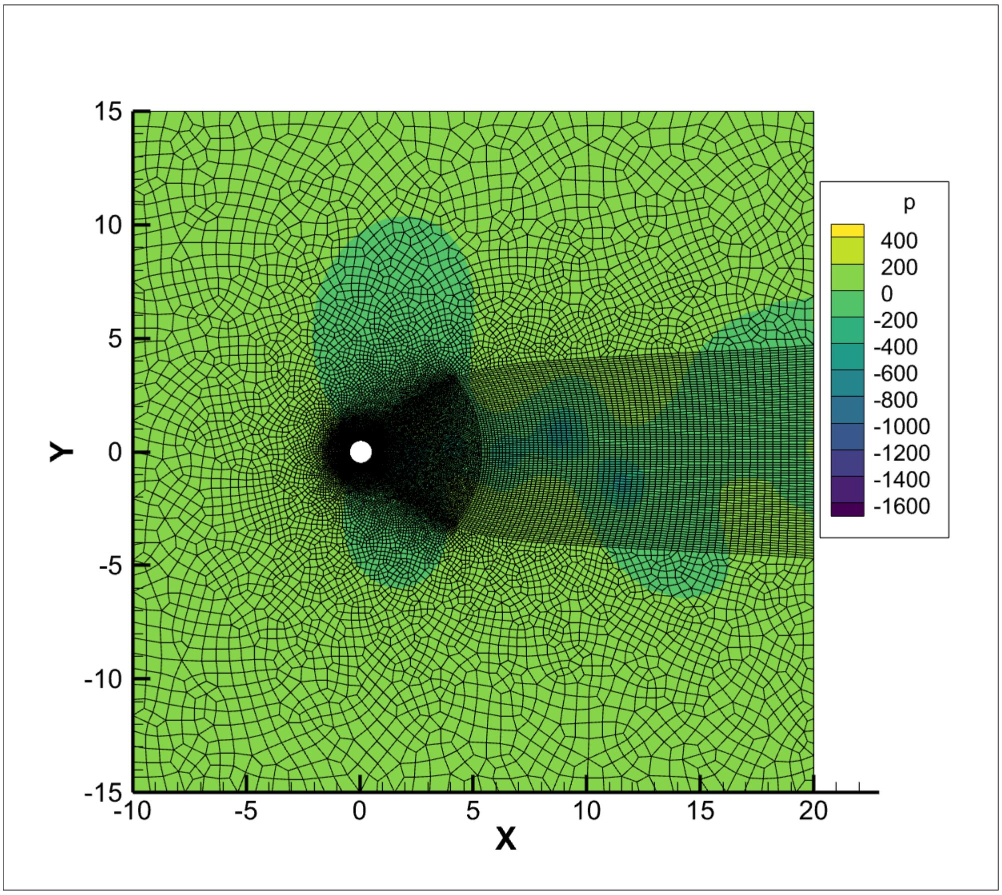


Figure 5: Mesh and Contour at 41 second

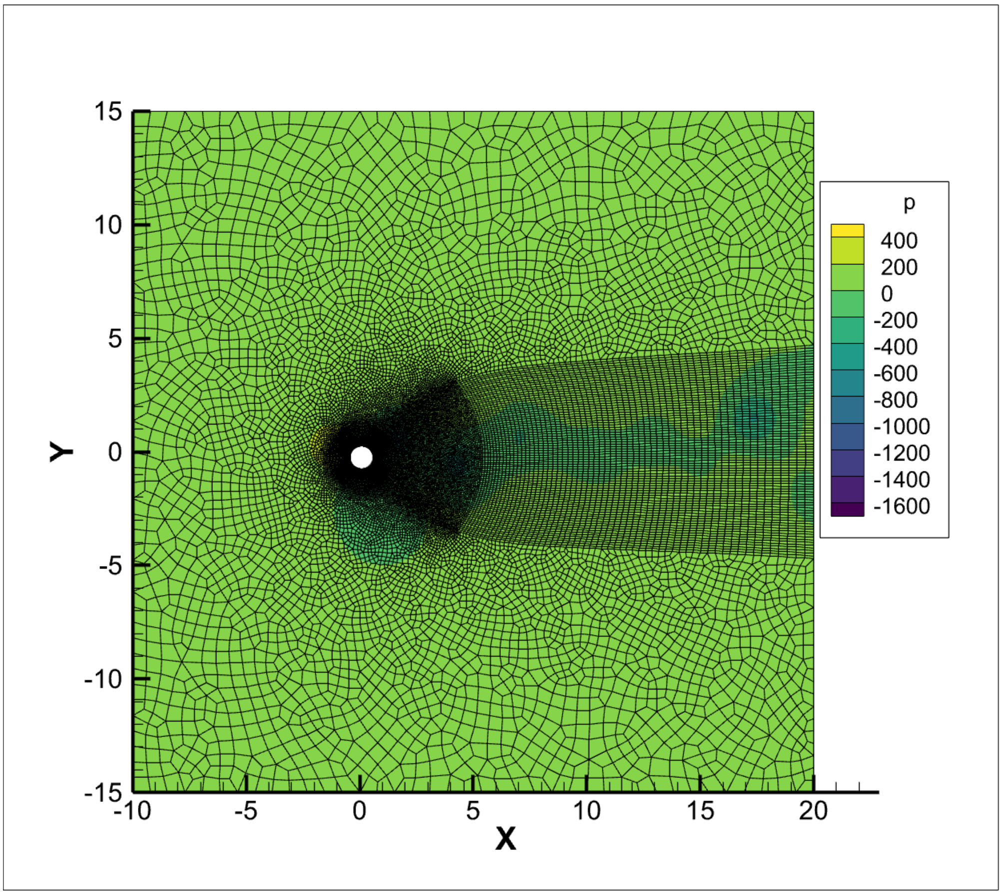


Figure 6: Mesh and Contour at 51 second

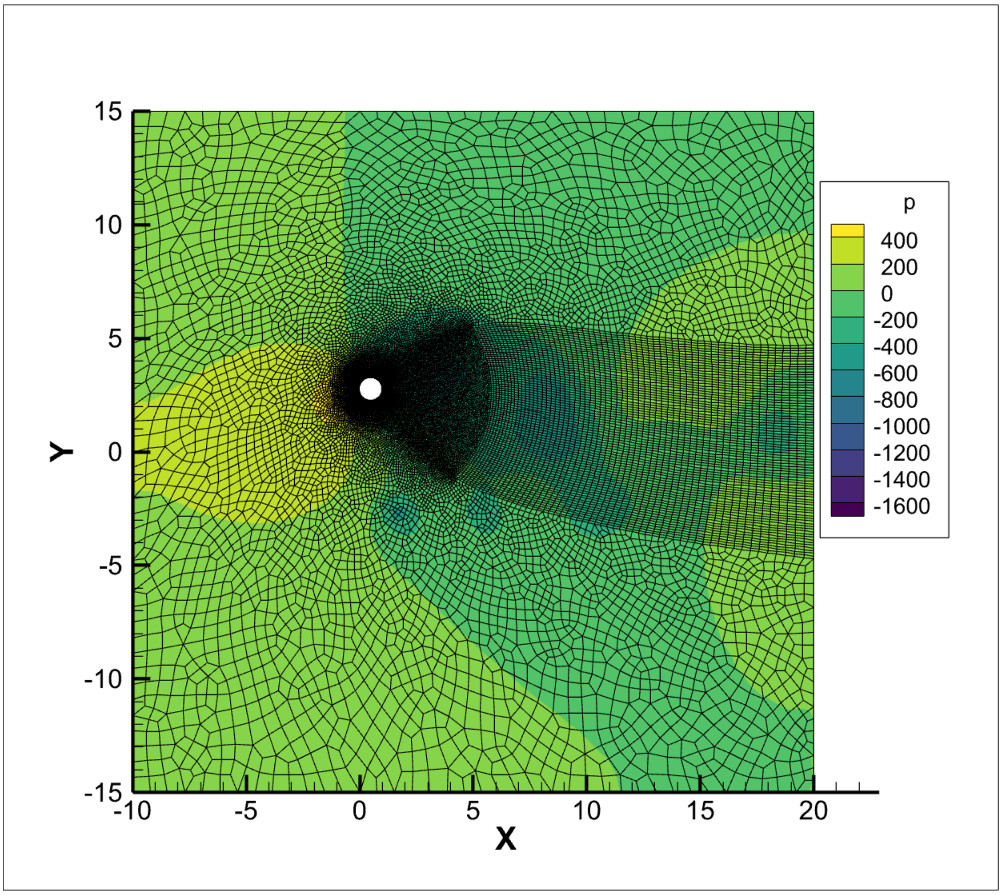


Figure 7: Mesh and Contour at 102 second

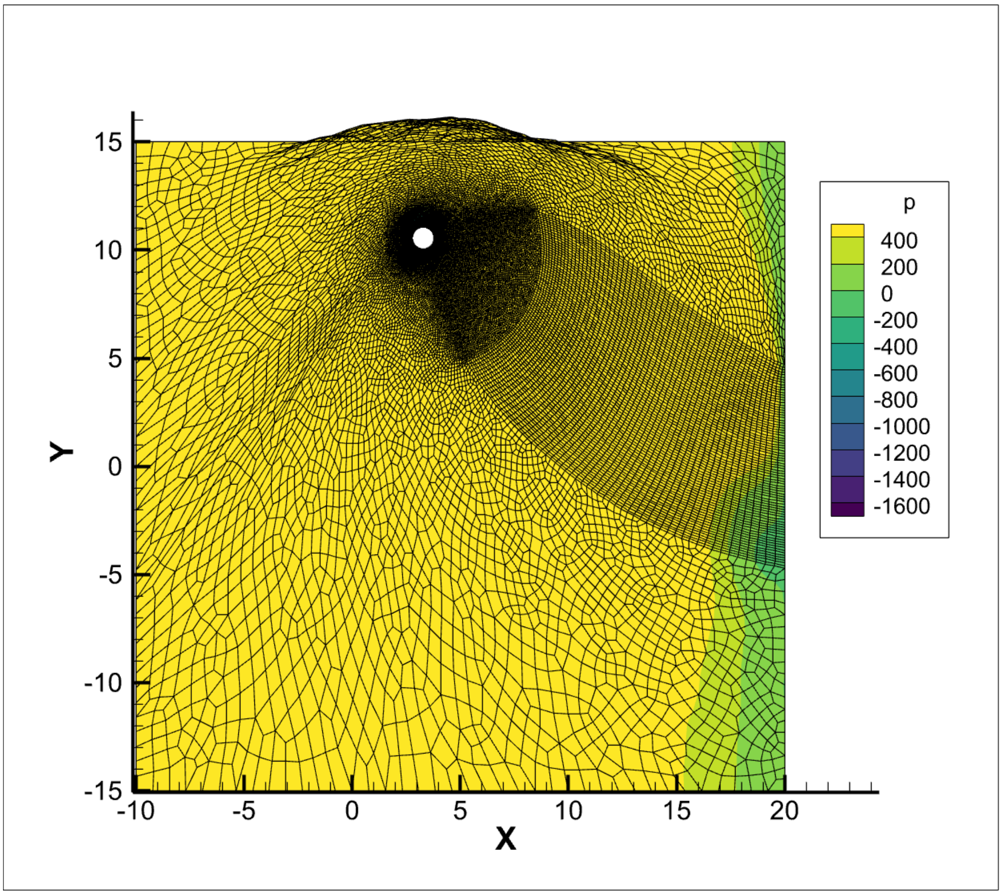


Figure 8: Mesh and Contour at 153 second

# Conclusion

This project aimed to deepen the understanding of fluid-structure interactions through the study of vortex-induced vibrations in a rigid circular cylinder. Through the successful implementation of the IntegratedOutput function, the project achieved an accurate calculation of forces acting on the cylinder, validated against reference values. This accomplishment underscores the robustness of the employed methodologies in modeling the fluid dynamics aspect of FSI problems. However, the endeavor to accurately simulate the ALE mesh for the representation of mesh movement encountered challenges. The discrepancies observed in the simulation of vortex-induced vibration scenarios highlight the complexities involved in the ALE mesh implementation and the sensitivity of FSI simulations to accurate mesh movement modeling.

The project's exploration into the realm of fluid-structure interactions has shed light on the intricate dance between fluid dynamics and structural responses. The partial success in achieving the project's objectives illustrates the nuanced understanding required to navigate the FSI domain and the importance of precision in implementing computational models. Future work should focus on refining the ALE mesh implementation, exploring the impacts of different mesh movement strategies, and enhancing the robustness of the simulation framework to better capture the complexities of real-world FSI scenarios. Through continued investigation and refinement, the field can advance towards more accurate and reliable modeling of fluid-structure interactions, with wide-ranging applications in engineering and beyond.

# Reference

[1]:"On the vortex-induced oscillations of a freely vibrating cylinder in the vicinity of a stationary plane wall" L. Zhong, W. Yao, K. Yag, R. Jaiman, B. C. Khoo, Journal of Fluids and Structures, 65, Pg. 495-526 (2016).