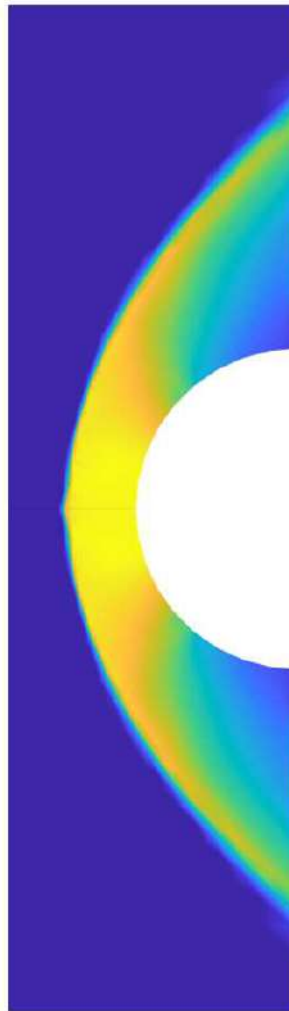# Development of Adaptive Remeshing Software for Compressible Fluid Flow Analysis

Bow shock from high-speed flow over cylinder
using adaptive mesh refinement

Dit Dejphachon

Thanapol Supitayakul

Engineering Project Final Report

Development of Adaptive Remeshing Software for Compressible Fluid Flow Analysis

Submitted by

Dit Dejphachon	6131765321
Thanapol Supitayakul	6131776221

Approved by:

Project Advisor : Professor Pramote Dechaumphai

Project Committee Member : Assoc. Professor Dr. Ekachai Chaichanasiri

Project Committee Member : Assoc. Professor Dr. Niphon Wansophark

Acknowledgement

We would like to express our gratitude and appreciation to those who gave us a possibility to complete this project. An utmost gratitude is given to our project advisor, Professor Pramote Dechaumphai, who gave us various important suggestions that ultimately lead to the success of this project by guiding on how to properly conduct both experiment and report. Moreover, we would like to thanks the project committee members, Assoc. Professor Dr. Ekachai Chaichanasiri and Assoc. Professor Dr. Niphon Wansophark, for their insight for the project and suggestions on how to improve the project.

Abstract

This report presents the design, documentation, and a user guideline of adaptive meshing software. The developed software is created to automatically improve the quality of the mesh based on previous flow behavior results calculated from the program hiflow . The aim of this project is to redevelop a software based on the old Fortran program, REMESH , in a newer and more supported programming language, MATLAB, and add features to the program hiflow such that the user can easily display a new and improved mesh. The newly generated mesh will greatly increase the accuracy of the result of the problem. Moreover, the developed software will allow the future student to explore and visualize various complex flow interactions which will greatly help in their learning of the Aerodynamic II course. In this project, the program is used to solve an array of problems: flow over wedge, flow over cylinder, flow over a diamond-shape airfoil, and expansion flow. The results show that the calculated solution of the density distribution at certain section in the flow domain become more accurate (Approaching the exact solution) every time the adaptive meshing is applied. Moreover, with MATLAB's API [1], the solution can be displayed in form of plots and gradients which greatly makes the user experience more interactive and flawless.

Contents

List of Figures

List of Tables

1 Introduction

In aerospace engineering, the various design problems are limited by a flow interaction with complex geometry. For that reason, computational fluid dynamics or CFD software [2] is widely used by engineers to preliminary test their design by simulating the flow behavior without the need for uneconomical modeling and wind tunnel testing. The use of CFD also allows the engineer to understand the flow interaction with complex geometry in a supersonic regime that cannot be done using the classical method.

To help a student in their learning of the compressible flow behavior in 2145312 Aerodynamic II , Professor Pramote Dechaumphai. Ph.D., who is responsible for the course, uses the finite volume-based CFD program called hiflow [3] that he wrote which allows the user to easily visualize the flow behavior. However, the flow behavior greatly depends on the quality of the input mesh to the program. For example, an accurate representation of a shock wave from a supersonic flow requires a finer or smaller mesh size that results in higher computational time.

For that, an adaptive remeshing software is needed to automatically improve the mesh quality based on previous flow behavior results from hiflow . The goal of this project is to redevelop an old Fortran program, REMESH in a newer MATLAB language and add compatibility to hiflow such that the user can easily display a new and improved mesh. The new mesh will greatly increase the accuracy of the result of the problem. The redeveloped REMESH program will also allow the future student to explore and visualize various complex flow interactions which will greatly help in their learning of the Aerodynamic II course.

2 Background

In general, many CFD program utilized finite volume method[4]. The FVM discretized the partial differential equations into algebraic equation that can be easier to solve. [5]. A flow problem can be discretized and solved by recreating the computational domain as a mesh and iterate to find the solution. As such, the quality of the mesh is one the most essential factor in the accuracy of the result [6]. In an ideal world, a numerical simulation with a smaller mesh size may better represent the real-world phenomena. For example, a minuscule-size Eddie of a turbulent flow can be better simulated while still retaining the clarity of the bigger flow domain. However, with the decreasing mesh size of any given problem, it also results in a higher amount of mesh needed to be calculated as well. This fact has been a great obstacle in the field of computational physics for a long time. As the higher amount of element count would increase the computational time and the resource needed to calculate and store it. Fortunately, with the increasingly more powerful computer and software suite, engineers can now reliably simplify and solve much general flow problems with ease. Nevertheless, in a field that involve complex physics phenomena such as compressible flow problem, the size and number of element greatly matter [7].

There are many ways to solve this problem; a notable example would be a technique called "adaptive mesh refinement" [8]. Consider a uniform mesh shown in figure 1 where a supposedly oblique shockwave with an angle of 45°, a solution obtained through the finite volume method would not be that much accurate. Although the result may appear to look course, the computational time is relatively fast. To obtain a better result, we can try to decrease the element size as shown by the 4 by 4 grid mesh of the problem shown in figure 2. The accuracy of the shockwave is much more accurate from decreasing the element size. If we were to repeat diving each grid furthermore, we would continue to have a better result as well. However, it can be seen that there are regions of the problem where the change with respect to space is rather

2 x 2 Square Grid

Oblique Shockwave –
Exact Solution
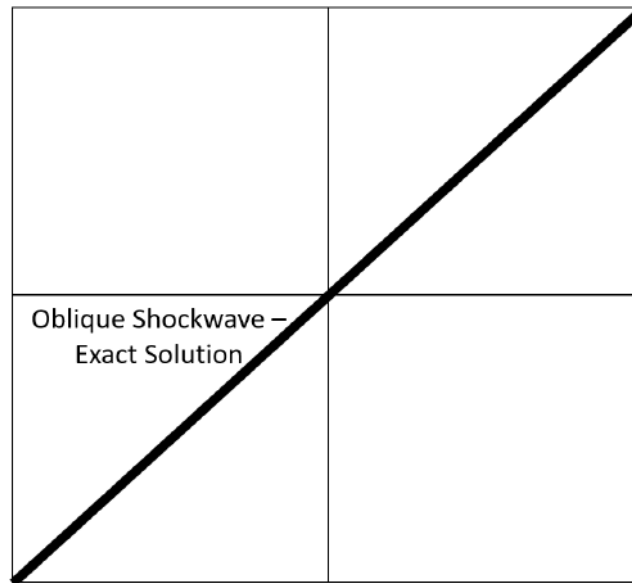
Figure 1: Oblique shockwave in 2 by 2 square grid

4 x 4 Square Grid

Oblique Shockwave –
Exact Solution
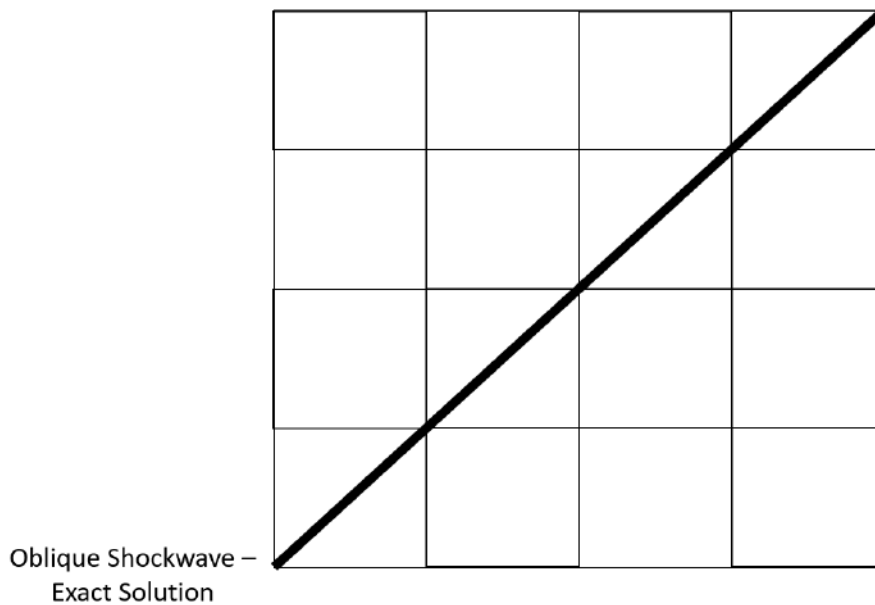
Figure 2: Oblique shockwave in 4 by 4 square grid

uniform. In this example, this would be a region that does not include the immediate surrounding of the shockwave line. As such, by repeatedly dividing these "regions" into a smaller mesh, we keep recalculating for the region with a rather uniform solution. Where each mesh and its immediate neighbor have similar flow properties. This would mean that our program would keep solving a region of small mesh with the same outcome, which would be unnecessary and a waste of computational resources. To counteract this, we can apply adaptive mesh refinement by instead of diving into every element, we only divide the mesh in which the change of element property is greater than what we are concerned with. For the oblique shockwave problem, this would mean the element that the shockwave directly lies on and its immediate neighbor as shown in figure 3. This technique allows us to capture the phenomena much better while keeping the number of element in check. Yet we can still see that the outer region far away from the

4 x 4 Square Grid Refined Mesh
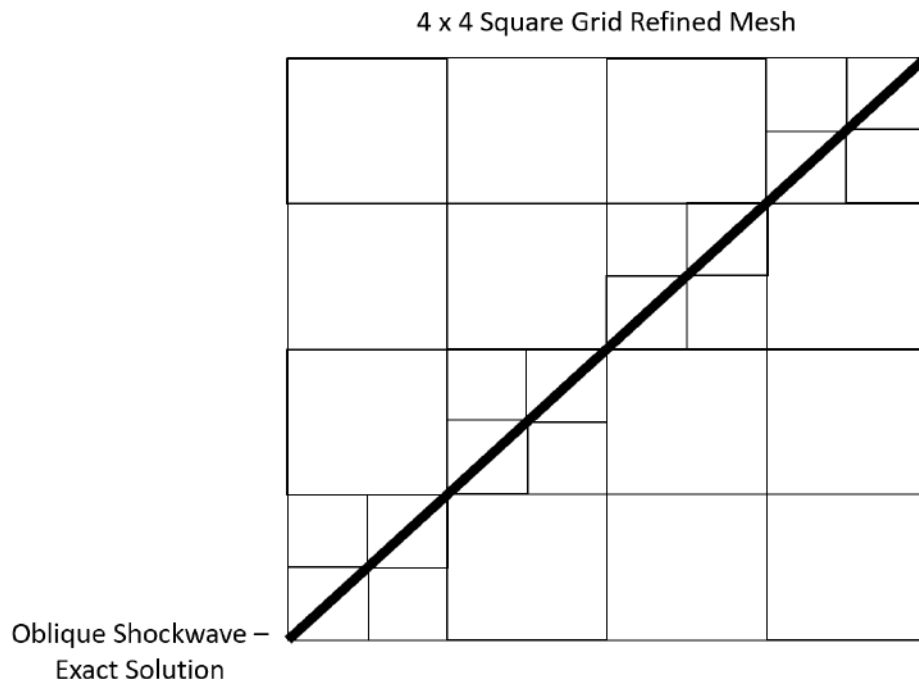
Figure 3: Oblique shockwave in 4 by 4 refined square grid

shockwave line is rather uniform. For this, we can merge these meshes in question altogether and create a smaller new element surrounding the shockwave line as shown in figure 4. This

4 x 4 Square Grid Remeshed
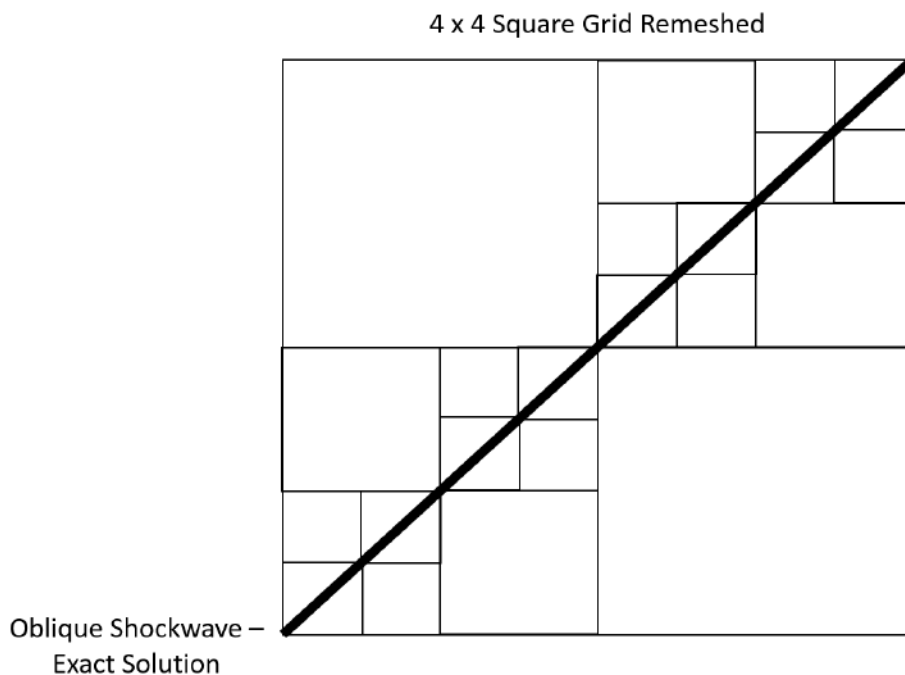
Figure 4: Oblique shockwave in 4 by 4 remeshed square grid

technique "remesh" solve the previous problem from adaptive mesh refinement by remapping the whole mesh based on the uniform mesh's solution. This allows the user to compute the solution through the uniform-large-size mesh first to identify the phenomenon before remeshing the problem in order to better capture that phenomenon. By concentrating a small-size element mesh onto the phenomena and larger-size to others region.

3 Design Requirements

Table 1: Design Requirements

No.	Design Requirements	Expected Value
1	Must be written in MATLAB language	true (1)
2	Must be operate via built-in CLI (Command Line Interface)	true (1)
3	Must be able to import/export calculation file	true (1)
4	Must be able to shown the newly generated mesh using plot function	true (1)
5	Must be able to integrate itself as a module to hiflow program	true (1)

Table 1 shows the design requirements for this project. Since the goal is to redevelop the original program written in Fortran language into MATLAB language , the first requirement is a must. The choice for the program to be operable through the built-in CLI [9] is for simplicity of development and integration into the existing hiflow program CLI. The third requirement is to ensure that the user can export the output to be used as input in another session. The fourth requirement is a feature that is not presented in the original Fortran program and added to the requirement. This feature is planned to be added such that the user can see the generated and enhanced mesh immediately. The last requirement is that the program must be able to integrate with the hiflow program. The integration to hiflow is very essential to ensure the flow of user experience. The redeveloped "remesh" program is software that takes the previous solution of the problem and creates a new mesh based on it.

4 Design Description

4.1 Overview

The program REMESH was initially written in Fortran. The function of this program is to resize the element size which increases the accuracy of the solution to the high-speed compressible flow problem. Consequently, the computation time is greatly reduced. The program's calculation steps can be summarized in figure 5.
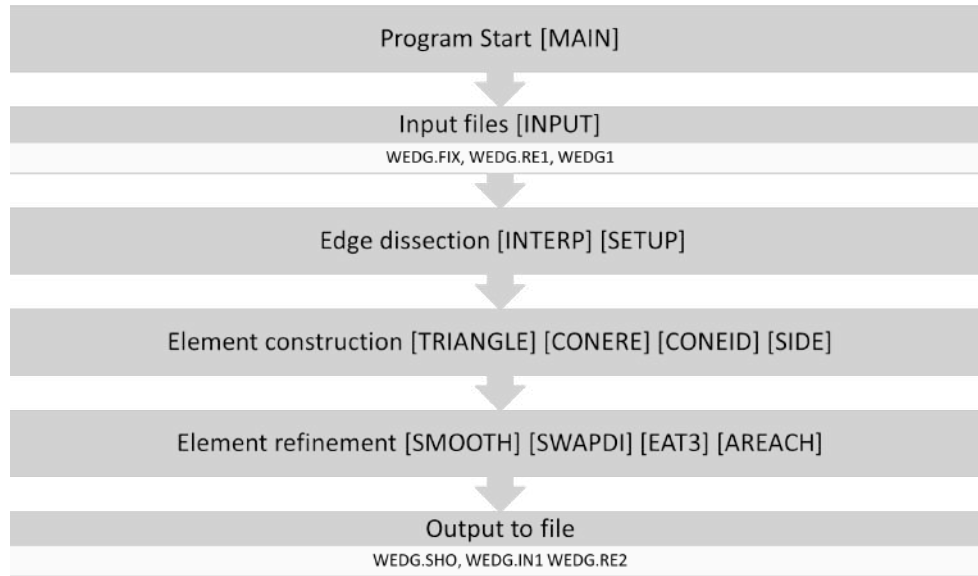
Figure 5: Calculation Steps of REMESH Program

From figure 5, the program consists of 5 main steps: input from files, edge dissection, element construction, element refinement, and output to file.

4.1.1 Input from Files

In this step, there are 2 cases in input files. The first case is inputting files for 1st adaptive mesh, and the second case is inputting files for n^{th} adaptive mesh. These 2 cases are operated by calling the INPUT subroutine.

4.1.2 Edge Dissection

The outer edges are divided into sections to generate elements by calling INTERP and SETUP subroutines.

4.1.3 Element Construction

The elements are generated within a boundary which occurs simultaneously with data storing used in the refinement process in the next step. This step is operated by calling TRIANGLE, CONERE, CONEID, and SIDE subroutines.

4.1.4 Element Refinement

This step is to refine the generated elements by calling SMOOTH, SWAPDI, EAT3, and AREACH subroutines.

4.1.5 Output to File

This step is to output the solutions by calling OUTPUT subroutine. Then, the output of the program will be used in the FINITE program.

Since this program will be used together with hiflow , the detailed technical explanation will be discussed in section 4.2.

4.2 Detailed Description

4.2.1 Input Files

There are 2 cases in input files. The first case is inputting files for 1st adaptive mesh, and the second case is inputting files for n^{th} adaptive mesh. These 2 cases have 3 files which are as follow:

4.2.1.1 WEDG.FIX The first input file type is .FIX . It stores the boundary condition of the problem which is the same for 1st and n^{th} adaptive mesh. The fully label .FIX example can be seen in the appendix 7.2. It consists of 5 parts:

Size of the boundary From table 7, the data in this part consists of a number of surfaces (NREG), joint (NFN), and boundary (NBCS) of the problem.

Table 2: Example of Size of the Boundary

NREG	NFN	NBCS
1	5	5

Joint of the boundary From table 3, the data in this part consists of a joint index and coordinate in the x and y axis.

Table 3: Example of Joint of the Boundary

COORDINATES		
1	0.0	0.0
2	0.4	0.0

The Upper edge of the boundary From table 4, in the second row, the data in this part consists of edge index, a number of edges, an ID of edge, and edge condition. In the third row, the data in this part consists of a number of points on the edges.

Table 4: Example of Upper edge of the boundary

BOUNDARY SEGMENTS			
1	2	0	3
2	2	-	-

The Surface of the boundary From table 5, in the second row, the data in this part consists of the index of the surface and the number of boundaries on the surface. In the third row, the data in this part consists of an index of all boundaries on the surface.

6

Table 5: Example of Surface of the boundary

REGION SPECIFICATION				
1	5			
2	2	3	4	5

Parameters for FINITE **program** From table 6, in the second row, the data in this part consists of specific heat ratio, tolerance, and the number of calculations. In the third-fourth row, the data in this part consists of density, velocity in the x and y-axis, and total energy.

Table 6: Parameters for FINITE program

GAMMA	EPSLAM	NTIME	
1.4	0.01	2000	
INITIAL CONDITION			
1.0	1.0	0.0	0.698412

4.2.1.2 WEDG.RE1 The second input file type is .RE1 . It stores the alignment of the element on the boundary of the problem. The fully label .RE1 example can be seen in the appendix 7.2. It can be divided into 2 parts:

Files for 1st **adaptive mesh** The data for 1st adaptive mesh consists of 4 subparts:

Size of the boundary From Table 7, in the second row, the data in this part consists of the number of joints, the element, initial joints on the wall, joints on the wall.

Table 7: Example of Size of the Boundary

NPOIG	NELEG	N1BODY	N2BODY
5	3	1	2

Data for Element Construction From Table 8, the data in this part consists of the index of joints, length of elements, the ratio of the length of the element, the first and second angle used in element construction, and coordinates on x and y axis.

Table 8: Example of Data for Element Construction

1	0.2	1.0	1.0	0.0	0.0	0.0

Joint Alignment Inside the Element From Table 9, the data in this part consists of an index of the element and the index of 3 joints.

Table 9: Example of Joint Alignment Inside the Element

1	1	2	5

Initial Condition From Table 10, in the second row, the data in this part consists of the index of joint, density, velocity in x and y axis, and total energy.

Table 10: Example of Initial Condition

INITIAL CONDITION			
1	1.0	1.0	0.0	0.698412

Files for n^{th} adaptive mesh   The data for n^{th} adaptive mesh consists of 4 subparts:

Size of the boundary   From Table 11, in the second row, the data in this part consists of the number of joints, elements, initial joints on the wall, joints on the wall.

Table 11: Example of Size of the Boundary for n^{th} adaptive mesh

NPOIG	NELEG	N1BODY	N2BODY
5	3	1	2

Data for automatic element resizing   From Table 12, the data in this part consists of the index of joints, length of elements, dimension of element, the first and second angle used in element construction, and coordinates in x and y axis.

Table 12: Example of Data for Automatic Element Resizing

1	1.0	1.0	0.0	0.698412	0.0	0.0

Joint alignment inside the element   From Table 8, the data in this part consists of an index of the element and the index of 3 joints.

Table 13: Example of Joint Alignment Inside the Element for n^{th} adaptive mesh

1	2	11	94

4.2.1.3   WEDG.G1   The third input file type is .G1 . It stores data used to dictate the flow of the program. It can be divided into 2 parts:

Files for 1st Finite Element   From Table 14, in the first row, the data consists of 1st finite element parameter which tells the program to execute 1st finite element program sequence which is set to 0 in the case. In the second row, the data consists of the ratio of boundary to the total fluid of the problem. In the third row, the data consists of a parameter that tells the program to reorder the element if the value is more than 1. In the fourth to thirteenth row, the data consists of mesh refinement parameters.

Table 14: Example of Files for 1st Finite Element

0
0.1
2
5
4
3
2
3
3
4
6
5
0

Files for Adaptive Mesh From Table 15, in the first row, the data consists of an adaptive mesh parameter which tells the program to execute an adaptive mesh program sequence which is set to 1 in the case. In the second row, the data consists of the length of the longest and shortest element and the length of the element that must be inspected. In the third row, the data consists of the dimension of the biggest element. In the fourth row, the data consists of a parameter that tells the program to resize the element which is set to 1 in this case. In the fifth row, the data consists of a parameter that tells the program to reorder the element if the value is more than 1. In the sixth to fifteenth row, the data consists of mesh refinement parameters.

Table 15: Example of Files for Adaptive Mesh

1		
0.08	0.03	0.06
3.0		
1		
2		
5		
4		
3		
2		
3		
3		
4		
6		
5		
0		

4.2.2 Output Files

The output file has a file format of ProblemName.out . The output file consists of Nodal density, velocity,and total energy. For example, the output for compressible flow over wedge problems for the 1st and 3rd adaptive mesh can be seen in tables 16 and 17, respectively.

Table 16: wedg.out for 1st Adaptive Mesh

Nodal density, velocity and total energy solutions [344]:			
Node	density	u-velocity	v-velocity	total energy
1	1.000000E+00	1.000000E+00	0.000000E+00	8.000000E-01
2	1.149762E+00	9.336493E-01	5.572766E-02	7.835954E-01
3	1.000000E+00	1.000000E+00	1.179409E-16	8.000000E-01
4	1.000000E+00	1.000000E+00	1.829466E-16	8.000000E-01
5	1.000000E+00	1.000000E+00	7.896546E-17	8.000000E-01
6	1.000000E+00	1.000000E+00	9.266172E-17	8.000000E-01
7	1.000000E+00	1.000000E+00	9.256610E-17	8.000000E-01
8	1.000000E+00	1.000000E+00	9.730460E-17	8.000000E-01
9	1.000000E+00	1.000000E+00	7.825864E-17	8.000000E-01
10	1.000000E+00	1.000000E+00	2.826517E-17	8.000000E-01
		.		
		.		
		.		
915	1.000000E+00	1.000000E+00	4.058558E-11	8.000000E-01
916	1.000000E+00	1.000000E+00	3.937249E-10	8.000000E-01
917	1.000000E+00	1.000000E+00	2.567888E-09	8.000000E-01
918	1.000000E+00	1.000000E+00	9.128436E-09	8.000000E-01
919	1.000000E+00	1.000000E+00	3.692017E-17	8.000000E-01
920	1.000000E+00	1.000000E+00	2.252001E-17	8.000000E-01
921	1.000000E+00	1.000000E+00	1.686368E-16	8.000000E-01
922	1.000000E+00	1.000000E+00	-3.293900E-18	8.000000E-01

Table 17: wedg.out for 3rd Adaptive Mesh

| Nodal density, velocity and total energy solutions [922]:			
Node	density	u-velocity	v-velocity	total energy
1	1.000000e+00	1.000000e+00	0.000000e+00	8.000000e-01
2	1.271501e+00	8.787941e-01	8.603110e-02	7.709908e-01
3	1.000000e+00	1.000000e+00	-1.823739e-17	8.000000e-01
4	1.000000e+00	1.000000e+00	-4.287978e-17	8.000000e-01
5	1.000000e+00	1.000000e+00	-6.710151e-17	8.000000e-01
6	1.000000e+00	1.000000e+00	-2.693764e-17	8.000000e-01
7	1.000000e+00	1.000000e+00	2.836307e-17	8.000000e-01
8	1.000000e+00	1.000000e+00	-3.150832e-17	8.000000e-01
9	1.000000e+00	1.000000e+00	-6.076537e-17	8.000000e-01
10	2.145603e+00	7.346143e-01	2.673815e-01	7.463369e-01
		.		
		.		
		.		
337	1.000000E+00	1.000000E+00	-4.032889e-17	8.000000E-01
338	1.000000E+00	1.000000E+00	-8.053960e-17	8.000000E-01
339	1.000000E+00	1.000000E+00	-3.349954e-17	8.000000E-01
340	1.000000E+00	1.000000E+00	-9.925456e-17	8.000000E-01
341	1.000000E+00	1.000000E+00	-6.379487e-17	8.000000E-01
342	1.000000E+00	1.000000E+00	-1.296530e-17	8.000000E-01
343	1.000000E+00	1.000000E+00	5.050313e-17	8.000000E-01
344	NaN	NaN	NaN	NaN

4.2.3 Program Flow Chart
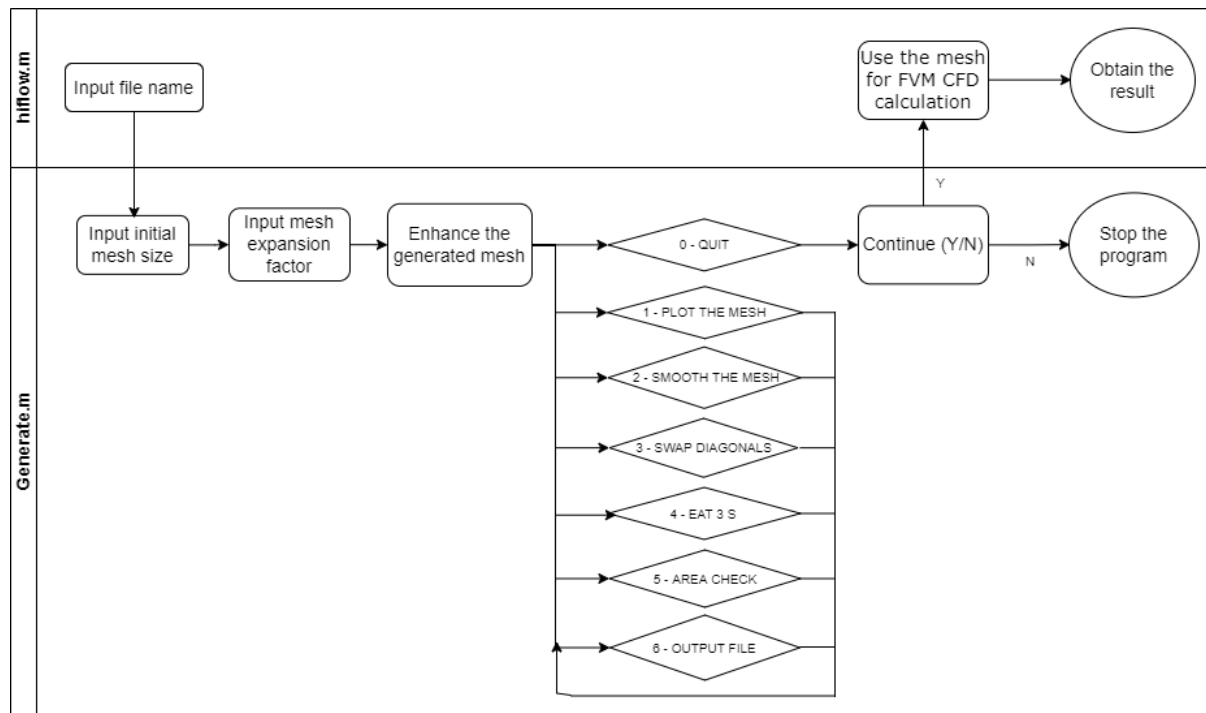
Figure 6: Block Diagram for Generation of Initial Mesh

The flow chart of hiflow and Generate program is shown in figure 6. Starting at hiflow section, the user will be asked to input the name of the problem that the user wants to solve. Moreover, .FIX , .RE1 , and .G1 files are required to run the program. These three files are utilized to create the initial mesh of the problem. On the other hand, the program can also take an existing mesh and problem of .DAT file as well.

Once the input files are ready, the hiflow is ready for execution. Moving on the Generate.m , the user will be asked to input the initial mesh size and expansion factor. These two parameters typically have value ranging from 0.03 - 0.05 and 0.1 for initial mesh size and expansion factor, respectively. Then, the uniform mesh of the problem is ready to be used for further calculation. Nonetheless, the mesh can be further enhanced for better calculation stability and accuracy.

Next, the user will be presented with 7 choices: Quit, Plot the mesh, Smooth the mesh, Swap Diagonals, EAT 3 S, Area check, and Output file. These choices can be executed by inputting a number starting from 0 to 6 into the console. As stated earlier, the mesh can be enhanced for a better accuracy and lessen the chance of diverging solution by using the 2 - Smooth , 3 – Swap diagonals , and 4 – Eat 3's functions. 2 - Smooth is used to tell the program to smooth the overall mesh for the number of times that the user has inputted. 3 – Swap diagonals is used to tell the program to swap the cell's diagonal lines. 4 – Eat 3's is used to tell the program to eliminate the point where the three elements coincide which can be also viewed as a triangle.

At this point, the generated initial mesh and important parameters are outputted as .DAT file to be used in further calculation by choosing (6 – Get the re-start file) and (0 – Quit), respectively. For the next section, the user can choose to input the character "y" or "Y" to the MATLAB console [10] to continue the calculation and "n" or "N" to stop the program.

Then, the process will be back to hiflow program. The program will continue the calculation using information based on the problem definition and previously generated mesh. Once the calculation has started, the animation graph will pop up showing the calculation residual where the user can stop by pressing 'f' and add more iterations when the graph reaches its limit.

Finally, the program has reached the output and result displaying section. The user will be asked to input the name of the solution file. The recommended format of the output file is `ProblemName.out`. Then, the mesh is taken to `hiflow.m` program to run a calculation using the Finite volume method based CFD program hiflow. Next, the user can choose what information he/she wants to plot. The options for the plot consist of mesh plot, density fringe plot, velocity vector plot, pressure fringe plot, and residual plot.
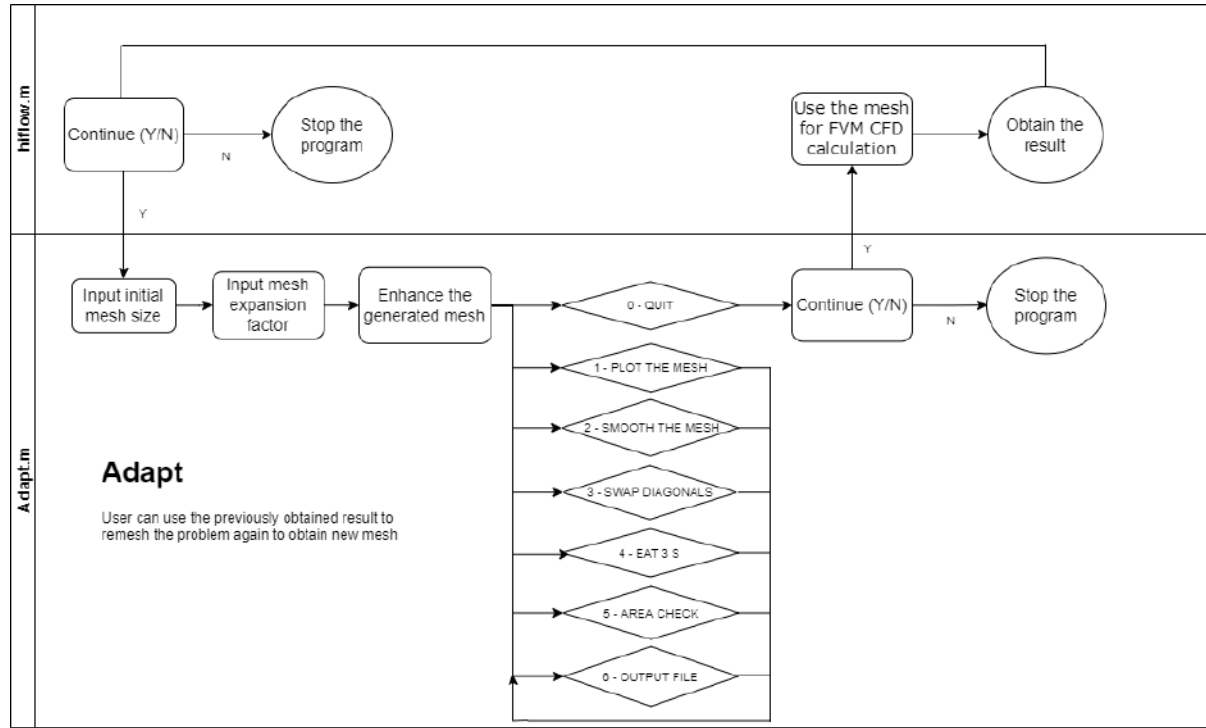

Figure 7: Block diagram for Generation of Adaptive Mesh

After the user is done with the generation of the initial mesh, the user will be presented with the option: continue and stop. If the user chose to continue the program will continue the calculation in `Adapt.m` which is a program dedicated to adaptive mesh generation. The solution file `ProblemName.RE2` generated from the previous calculation will be utilized as base information for further remeshing.

From Figure 7, in the 'input initial mesh size' block, the user will be asked to input element size information: maximum, minimum, and checking element size. These element sizes will be used as a basis for the calculation. Moreover, the user must specify the cell's maximum aspect ratio. Then, the user will be presented with a selection of key variables that will be used as a base for further adaptive mesh. The key variables are density, velocity (Modulus), density and velocity, entropy, density and Mach number, and Mach number. The selection of these keys is totally problem-related, so the user must have some idea about what key will yield the most stable and accurate calculation. Among these variables, option 4 to 6 (entropy, density and Mach number, and Mach number) must be specified with a specific heat ratio before the user can proceed.

In the 'input expansion factor' block, the user must input the expansion factor before continuing.

Next, in the 'Enhance the generated mesh' block, the user will be presented with 7 choices: Quit, Plot the mesh, Smooth the mesh, Swap Diagonals, EAT 3 S, Area check, and Output file. These choices can be executed by inputting a number starting from 0 to 6 into the console. The stage is the same as that of the `Generate.m` where the user has the ability to optimize the mesh to help obtain a more accurate and stable solution. Once the user is satisfied with the mesh, the program will be asked whether the user wants to continue the calculation or not.

12

If the user wishes to continue, the program will take the mesh information to `hiflow.m` to run a further calculations. In this stage, the calculation will be the same for any n^{th} version of the mesh. Finally, the user can iterate the process as many times as he/she please.

4.3 Use Cases

4.3.1 Flow impinging on wedge

Using the process stated in the section 4.2, the program can be used to solve various aerodynamics problem. In this example, the compressible flow over wedge problems will be used to demonstrated how to use the program.

From figure 8, oblique shock wave is generated from an impingement of a Mach 1.6 flow at an angle of 20°.
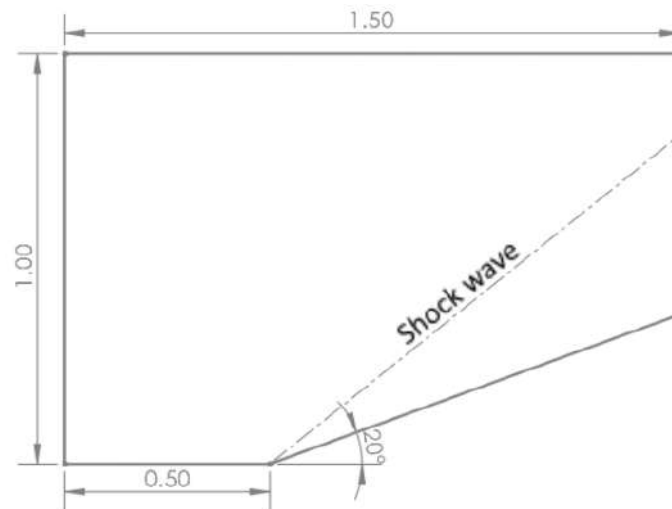
Figure 8: Flow over Wedge Problems

This problem and necessary information are specified in `.FIX`, `.RE1`, and `.G1` files which will be used as the input for the finite volume method program. These three files are used in generation of the initial mesh of the problem whereas the program can also take an existing mesh and problem of `.DAT` file as well.

4.3.1.1 Input File Generation   The user first need to translate all the problem dimensions into the necessary format. We will start with `.FIX` format where the first two lines of text contain the information about the boundary conditions. The flow domain of this flow over wedge problem has 5 edge which are all boundary conditions edge. The first two line of the `.FIX` file would be written as shown in figure 9

Figure 9: Boundary conditions information

The first parameter `NREG` is always set to 1 so the user should only type the number of boundary conditions under both `NFN` and `NBCS`. In this example,the dimension shown in figure 8 can be rewritten as a series of vertices on line 3 forward. These vertices will form the outline shape of the flow domain and are written in the format shown in figure 16. The first

Figure 10: Geometry Vertices information

column is the id of the coordinate starting from 1. The second and third column are vertices coordinate in x and y axis, respectively. Now, we need to give an information of how the vertices are connected to each other to form a properly flow domain. An example of this boundary segment section is shown on figure 11 With the first row starting at line 10 here and skipping

Figure 11: Boundary Segment Information

every other row downward, contain from left to right an edge index, number of edge, ID of edge, and edge condition. For most problem, user only need to specify edge index which start from 1 and edge condition. The value of 1 dictates inflow condition, 2 dictates outflow condition, and 3 being wall. The second line and every other line forward contain information on the vertices that make up this edge. For the last three sections of the file, the user needs to specify the region specification, which is in essence based on the edge number. Figure 12 shows an example of this problem. Under the section REGION SPECIFICATION , the user simply needs to enter 1 and the total edge count on the first line. Then enter an increment from 1 to the total edge count on the second line. The rest are as discussed earlier in section 4.2.1.1. The full example of the .FIX file used here in this problem is shown in figure 13.

```
20   REGION SPECIFICATION
21   1      5
22   1      2      3      4      5
23   INPUT GAMMA,EPSLAM,NTIME
24    1.40         0.01        1000
25   INITIAL CONDITIONS
26    1.         1.         0.         0.8
```

Figure 12: Region and Initials Condition Information

```
≡ WEDG.FIX
1     NREG      NFN       NBCS
2     1         5         5
3     COORDINATES
4     1         0.        0.
5     2         0.5       0.
6     3         1.5       0.36397
7     4         1.5       1.
8     5         0.        1.
9     BOUNDARY SEGMENTS
10    1      2      0      3
11       1      2
12    2      2      0      3
13       2      3
14    3      2      0      2
15       3      4
16    4      2      0      1
17       4      5
18    5      2      0      1
19       5      1
20    REGION SPECIFICATION
21    1      5
22    1      2      3      4      5
23    INPUT GAMMA,EPSLAM,NTIME
24     1.40         0.01        1000
25    INITIAL CONDITIONS
26     1.         1.         0.         0.8
```

Figure 13: Example of .FIX file for flow over wedge problem

For .RE file, the first two lines contain the number of points, the number of initial mesh elements, the number of initial joints on the wall and the joints on the wall.

NPOIG	NELEG	N1BODY	N2BODY
5	3	1	2

Figure 14: .RE file first section

Figure 14 show an example of this file for this problem. Again, the first column is the number of vertices. Here, the second column is the number of initial elements. The initial element in this case as shown in figure 15. .FIX The flow domain are divided into three triangular region
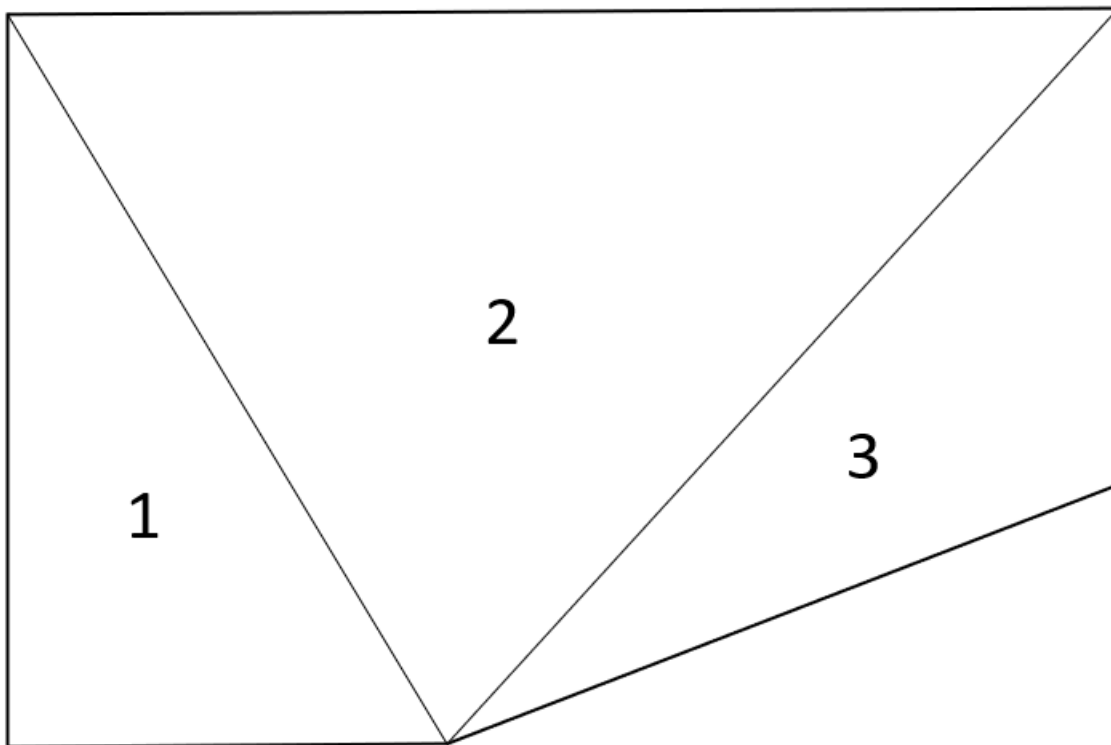
Figure 15: Initial Element of the Problem

or three initial elements. This is done so that the remesh program can generate a uniform initial mesh from it. Which translate to the value of 3 under NELEG section, whereas N1BODY is always set to 1 while N2BODY is the number of wall boundary condition. The coordinate sections are as discussed earlier in Section 4.2.1.2.

3	1	0.03	1.	1.	0.	0.	0.
4	2	0.03	1.	1.	0.	0.5	0.
5	3	0.03	1.	1.	0.	1.5	0.36397
6	4	0.03	1.	1.	0.	1.5	1.
7	5	0.03	1.	1.	0.	0.	1.

Figure 16: .RE file coordinate section

Figure 17 shows an element connectivity section. The user can specify the initial triangular element by giving a three points that make up a single element. The first column is the triangular

element id and the rest of the column column are the points or vertices id that makes up an element. Noted that while specifying the points, it has to be in an increasing order.

Figure 17: .RE file element connectivity section

Figure 18 shows the initial conditions section of the file. The format and input here are as discussed earlier where the user need to specify the four unknown variable at each points id. The first column again is a points id with the rest being density, x-velocity, y-velocity, and energy, respectively.

Figure 18: .RE file initial condition section

However, the user do not actually have to perfectly divide within the flow domain as alternative method shown in figure 19. By creating a hypothetical point that does not exist in the flow domain and thereby allowing us to divide the flow domain into two equal triangular elements. This method is advised for a complex geometry problem that cannot easily created the initial triangular element from it. Still, this method has not been tested under different problems, and the user should be aware that there might be an unexpected problem that might came up from using it. An example of this method will be discussed in the use cases of flow over cylinder in later section.
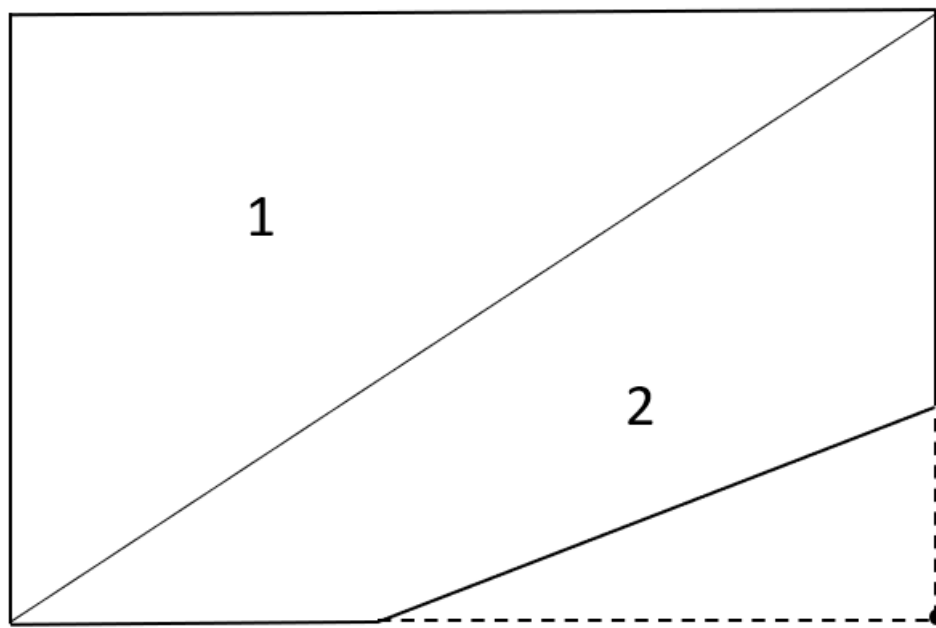
Hypothetical point

Figure 19: Initial Element of the Problem - Alternative Method

The hiflow program is then executed as specified earlier. Using an initial uniform mesh size of 0.04 and expansion factor of 0.1, the mesh generated from Generate.m function is as shown below.

```
1   Hiflow
2
3   .(d)at or .(f)ix data input?: f
4   Enter the input geometry file name: wedg
5   -------------------------------------------------
6    *** READING DATA
7   wedg.RE1
8   Input initial uniform mesh size
9   0.04
10
11  *** FILLING IN THE GAPS
12  *** FILLING ISIDE FOR THE BACKGROUND GRID
13  *** FILLING ELEMENT CONECTIVITY MATRIX
14  Input expansion factor (generally close to 0) :
15  0.1
16  AE =
17      0.5000
18
19   *** GENERATING BOUNDARY NODES
20   *** INTERPOLATING FROM THE BACKGROUND GRID
21  aw for re-ordering, (aw=1.:always,aw=large:never)
22      2
23
24   *** GENERATING BOUNDARY NODES
25   *** WHAT SHALL WE DO NOW ?
26      0 - QUIT
27      1 - PLOT THE MESH
28      2 - SMOOTH THE MESH
29      3 - SWAP DIAGONALS
30      4 - EAT 3 S
31      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
32      6 - GET THE RE-START FILE
33   Please select from the menu :
```

At this stage, the uniform mesh of the problem is ready to be use for further calculation as shown in figure 20. Nonetheless, the mesh can be further enhanced for better calculation stability and accuracy. This is obtained by smoothing the overall mesh (2 - Smooth), swapping the cell's diagonal line (3 – Swap diagonals), and removing the point where three element coincide (4 – Eat 3 s). Using a smoothing loop of 5 and swapping the diagonal together with removing coincide point once, the enhanced mesh can be seen below.

Figure 20: Flow over Wedge Problems Uniform Mesh

```
*** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
2
     2

ENTER NUMBER OF SMOOTHING LOOPS
5
 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
3
     3

27.000000 sides have been swapped
0.000000 sides have been swapped
filling iside
 *** WHAT SHALL WE DO NOW ?
```

20

```
        0 - QUIT
        1 - PLOT THE MESH
        2 - SMOOTH THE MESH
        3 - SWAP DIAGONALS
        4 - EAT 3 S
        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
        6 - GET THE RE-START FILE
 Please select from the menu :
4
        4

nr. of 3's removed = 0 *** WHAT SHALL WE DO NOW ?
        0 - QUIT
        1 - PLOT THE MESH
        2 - SMOOTH THE MESH
        3 - SWAP DIAGONALS
        4 - EAT 3 S
        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
        6 - GET THE RE-START FILE
 Please select from the menu :
5
        5

The checking was successful
Total number of generated points : 922.000000
Total number of generated elements : 1722.000000
```
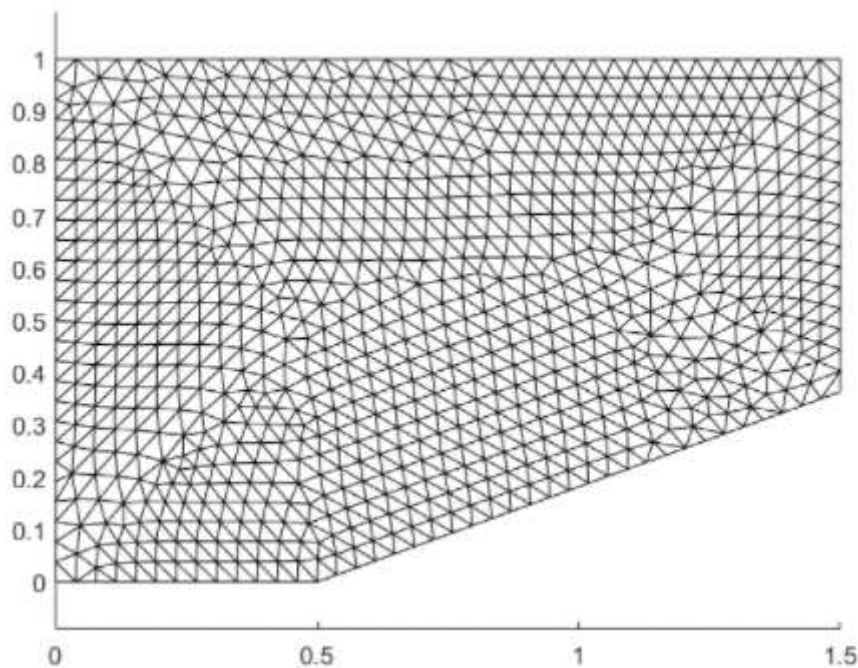

Figure 21: Flow over Wedge Problems Enhanced Mesh

Noted that the enhanced mesh in this example is relatively the same as the original one as shown in figure 21. As the enhance functions are intended for uses with an already adapted mesh which we will see it in greater effect later in this example. Now, the generated initial mesh and others problem information are outputted as .DAT file to be used in further calculation by invoking (6 – Get the re-start file) and (0 – Quit) in that order. To continue the calculation using the generated mesh, input character "y" or "Y" to the MATLAB console [10] and "n" or "N" to stop the program.

```
*** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
6
     6

 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
0
     0
stop
Continue:? (Y/N)
Y
```

The hiflow program will start calculating based on the problem definition and the mesh obtained earlier. A figure showing the current residual plot [11] will pop-up automatically for users to observe the real time updating of calculation residual as shown in figure 22. If the current residual values are satisfactory enough, simply press "f" to stop the calculation immediately.
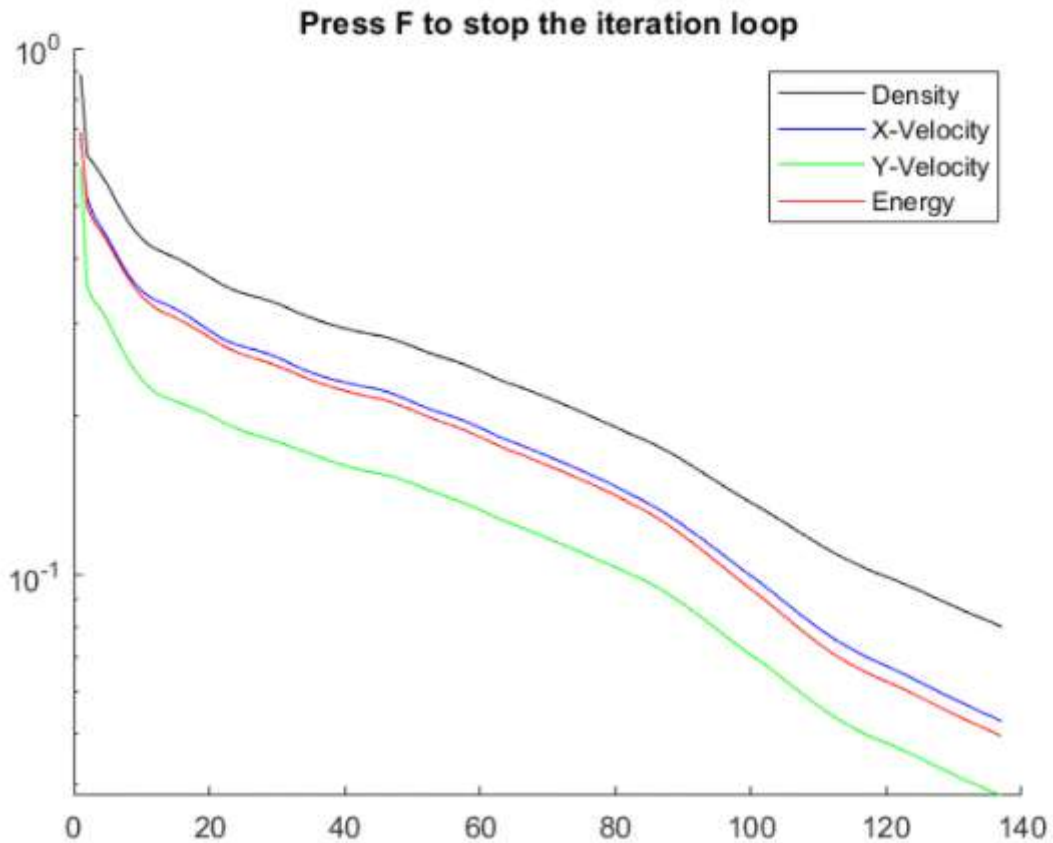
Figure 22: Flow over Wedge Problems Iteration

```
1   The finite element model consists of:
2       number of nodes                    =     922
3       number of elements                 =    1722
4       number of boundary conditions      =     120
5       number of iterations needed        =    1000
6
7    Performing iterations for convergence
8     Iter      Del rho    Del rho-u    Del rho-v    Del rho-e
9        1 8.91247e-01 6.75199e-01 5.93587e-01 6.93176e-01
10      10 4.35735e-01 3.46782e-01 2.35369e-01 3.38676e-01
11      20 3.67776e-01 2.91188e-01 2.01134e-01 2.82843e-01
12      30 3.27611e-01 2.59101e-01 1.79201e-01 2.49624e-01
13      40 2.94041e-01 2.32259e-01 1.61562e-01 2.23907e-01
14      50 2.71725e-01 2.13158e-01 1.49263e-01 2.05501e-01
15      60 2.43889e-01 1.90353e-01 1.33031e-01 1.82920e-01
16      70 2.16934e-01 1.68115e-01 1.17456e-01 1.61467e-01
17      80 1.90920e-01 1.47347e-01 1.03515e-01 1.41697e-01
18      90 1.65086e-01 1.24260e-01 8.81398e-02 1.18832e-01
19     100 1.37339e-01 9.97191e-02 7.05869e-02 9.40575e-02
20     110 1.14594e-01 7.92886e-02 5.64291e-02 7.41982e-02
21                              .
22                              .
23                              .
24     960 8.16222e-15 9.46949e-15 2.69877e-15 7.04183e-15
25     970 7.94178e-15 9.22688e-15 2.50287e-15 6.92356e-15
```

23

```
26      980 8.15542e-15 9.65703e-15 2.58348e-15 7.19335e-15
27      990 7.97431e-15 9.42121e-15 2.39213e-15 6.97499e-15
28     1000 8.18408e-15 9.57628e-15 2.51079e-15 7.19849e-15
```

In the event that the specified iteration amount is not enough to obtain the desired convergence value, additional calculation can be done by typing "y" or "Y" to continue the calculation. The program will prompt the user to specify how many iterations for the calculation by simply input the positive integer to continue.

```
1  Do you want to continue calculating? (Y/N) :
2  y
3  How many more iteration?
4  1000
5
6  niter =
7
8          2000
```

Once the user is satisfied with the calculation convergence, enter the name for the file solution.

```
1  Enter file name for the solutions: wedg.out
```

Press "y" or "Y" to view the mesh used in the calculation, "n" or "N" to not show and continue as shown in figure 23.

```
1  Would you like to see the mesh plot? y/n: y
```
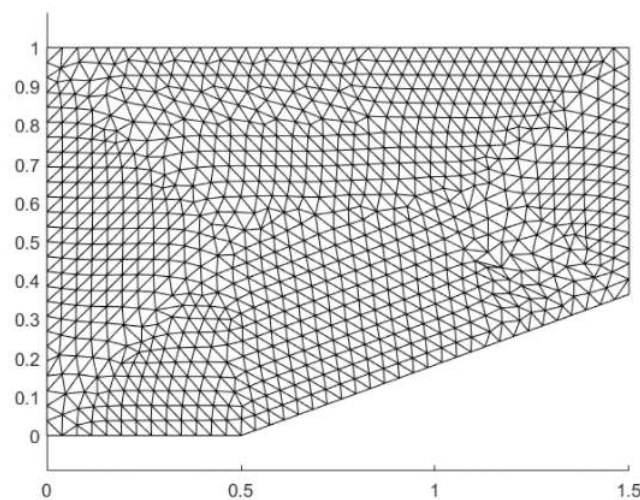

Figure 23: Flow over Wedge Problems Mesh Plot

Press "y" or "Y" to view the density fringe plot, "n" or "N" to not show and continue as shown in figure 24.

```
Would you like to see the density fringe plot? y/n: y
```


Figure 24: Flow over Wedge Problems Density Plot

Press "y" or "Y" to view the velocity vector plot, "n" or "N" to not show and continue as shown in figure 25.

```
Would you like to see the velocity vectors plotted on mesh? y/n: y
```


Figure 25: Flow over Wedge Problems Velocity Plot

Press "y" or "Y" to view the pressure plot, "n" or "N" to not show and continue as shown in figure 26.
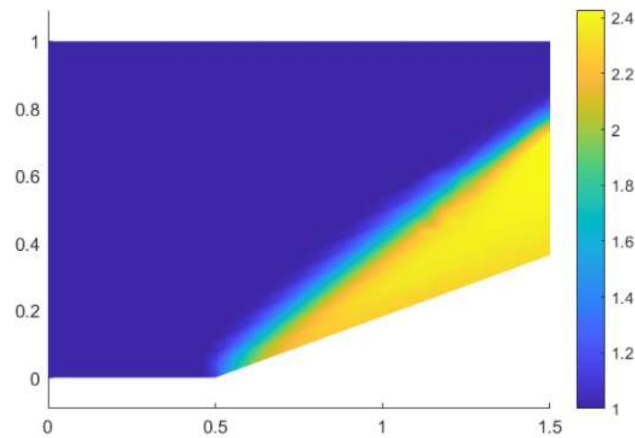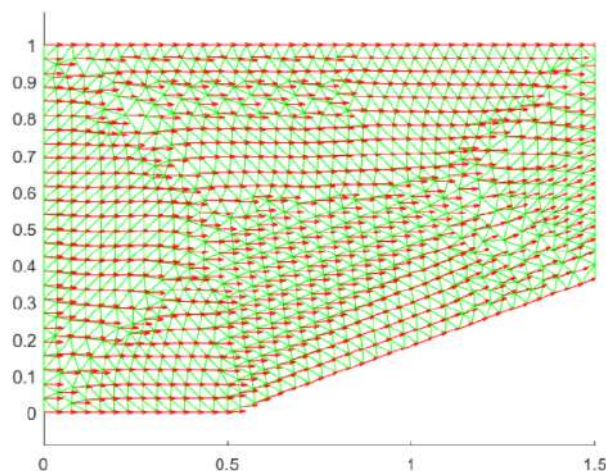
```
Would you like to see the pressure fringe plot? y/n: y
```
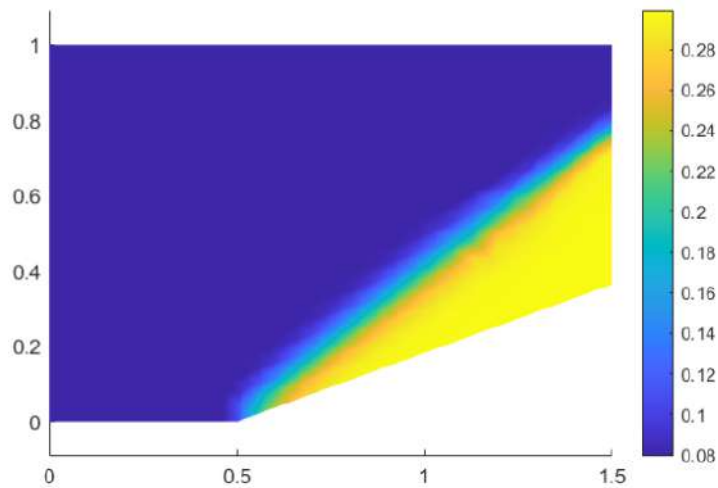
Figure 26: Flow over Wedge Problems Pressure Plot

Press "y" or "Y" to view the residual plot again, "n" or "N" to not show and continue as shown in figure 27.

```
Would you like to see the residual plot? y/n: y
```
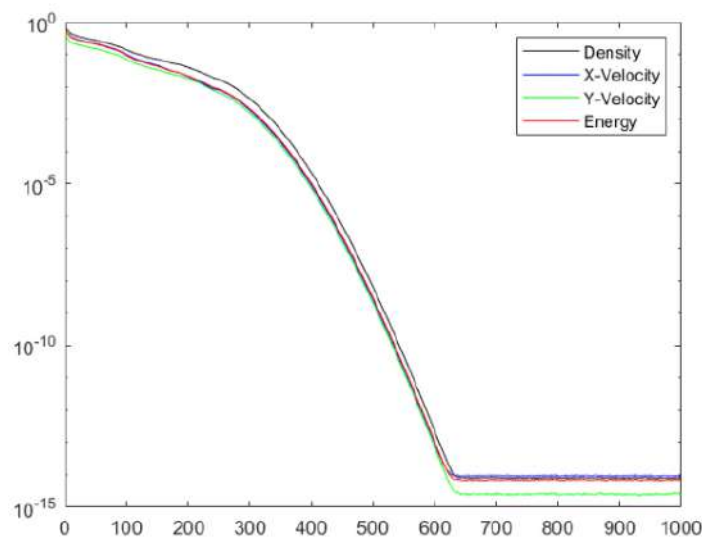

Figure 27: Flow over Wedge Problems Residual Plot

Press "y" or "Y" or apply adaptive remeshing, "n" or "N" to stop the program

```
Would you like to apply adaptive remeshing? y/n:y
------------------------------------------------
 *** READING DATA
wedg.RE2
```

The program will use the solution file (.RE2) obtained from the current calculation as a base for remeshing. The user must specify the maximum, minimum, and checking element size as a basis for the remeshing.

```
Input maximum element size
0.2
Input minimum element size
```

```
4    0.03
5    Input checking element size
6    0.03
```

Then, specifying the cell's maximum aspect ratio

```
1    Input cell aspect ratio
2    2
```

Now, selecting the key variable to be used as a base for adapting the mesh

```
1    WHAT DO YOU WANT TO USE AS KEY VARIABLE ?
2     1 - DENSITY
3     2 - VELOCITY (MODULUS)
4     3 - DENSITY + VELOCITY
5     4 - ENTROPY
6     5 - DENSITY + MACH NO
7     6 - MACH NUMBER
8    Please input the key variable for adaptive remeshing :
```

For option 4 through 6, user need to specify the specific heat ratio before continuing. In this example, using density and Mach number and specific heat ratio of 1.4 for the calculation.

```
1    Please input the key variable for adaptive remeshing :5
2         5
3
4     enter the value of gamma (usually 1.4)1.4
5    How many smoothing loops?
6    How many smoothing loops?
7     DO YOU WANT EXTRA REFINEMENT FOR STAGNATION POINTS ?
8         4
9
10   ENTER DELKM, THRESHOLD MACH NUMBER AND GAMMA
11        3     2     3
12
13    *** FILLING IN THE GAPS
14    *** FILLING ISIDE FOR THE BACKGROUND GRID
15    *** FILLING ELEMENT CONECTIVITY MATRIX
```

The expansion factor of 0.1 is used for this problem. Noted that this value should be close to 0 and that the variable AE and aw are pre-computed automatically by the program. So there is no need for the end user to input AE and aw variables.

```
1    Input expansion factor (generally close to 0) :
2    0.1
3    AE =
4        0.0385
5
6     *** GENERATING BOUNDARY NODES
7    Current progress in interpolation 20
8    Current progress in interpolation 40
9    Current progress in interpolation 6.000000e+01
10   Current progress in interpolation 80
```

```
11   Current progress in interpolation 100
12    *** INTERPOLATING FROM THE BACKGROUND GRID
13   aw for re-ordering, (aw=1.:always,aw=large:never)
14         3
15
16    *** GENERATING BOUNDARY NODES
17    *** WHAT SHALL WE DO NOW ?
18         0 - QUIT
19         1 - PLOT THE MESH
20         2 - SMOOTH THE MESH
21         3 - SWAP DIAGONALS
22         4 - EAT 3 S
23         5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
24         6 - GET THE RE-START FILE
25    Please select from the menu :
```
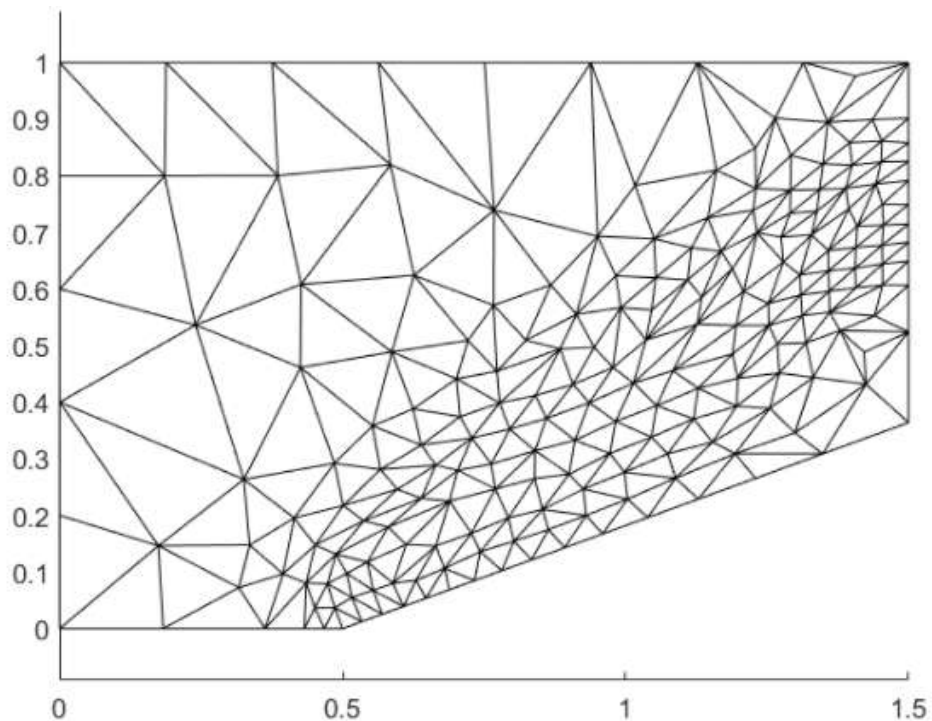

Figure 28: Flow over Wedge Problems 1st Adaptive Mesh

From figure 28, this result in a 1st adaptive mesh that are finer on the more sensitive shock line region and coarser in the free stream region. This mesh however still has some low-quality element that might hinder the stability and accuracy of the calculation. One of such is a long-thing cell element that is not align with the flow direction. As such, the mesh is enhanced first before attempting a calculation as shown below.

```
Please select from the menu : 2
     2

ENTER NUMBER OF SMOOTHING LOOPS
4
 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
3
     3

23.000000 sides have been swapped
1.000000 sides have been swapped
0.000000 sides have been swapped
filling iside
 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
4
     4

nr. of 3's removed = 0 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
     4 - EAT 3 S
     5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
     6 - GET THE RE-START FILE
 Please select from the menu :
5
     5
```

```
46   The checking was successful
47   Total number of generated points : 206.000000
48   Total number of generated elements : 366.000000
49   *** WHAT SHALL WE DO NOW ?
50        0 - QUIT
51        1 - PLOT THE MESH
52        2 - SMOOTH THE MESH
53        3 - SWAP DIAGONALS
54        4 - EAT 3 S
55        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
56        6 - GET THE RE-START FILE
57    Please select from the menu :
```
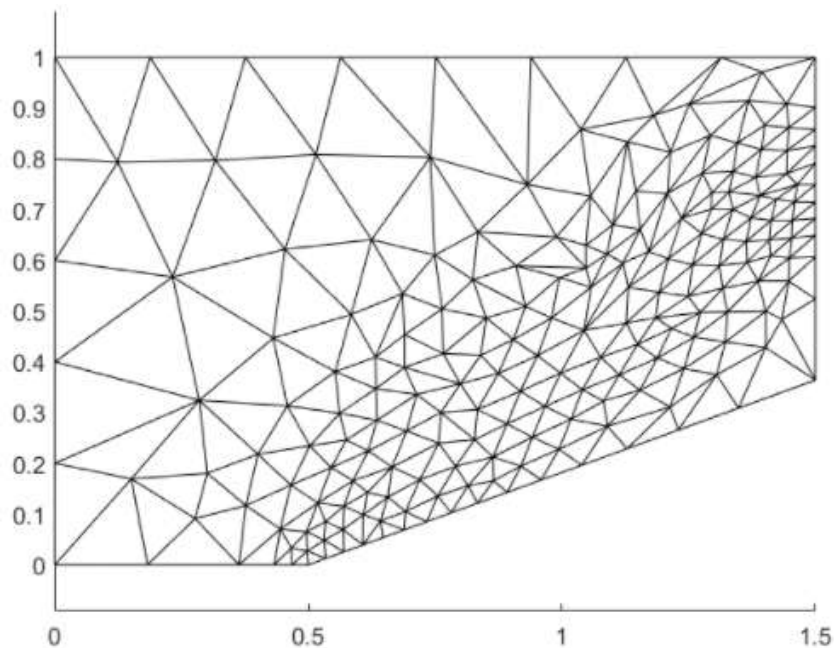

Figure 29: Flow over Wedge Problems Smoothed 1st Adaptive Mesh

By smoothing the element by 4 loop, swapping the necessary diagonal line, and tried EAT3 's function, the result mesh quality is now suitable for the calculation as shown in figure 29. The adapted mesh is then outputted together with the problem definition as a .DAT file for the hiflow program. This is done by selecting option 6 – GET THE RE-START FILE and 0 – QUIT in that order to generate .DAT file.

```
1    *** WHAT SHALL WE DO NOW ?
2         0 - QUIT
3         1 - PLOT THE MESH
4         2 - SMOOTH THE MESH
5         3 - SWAP DIAGONALS
6         4 - EAT 3 S
7         5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
8         6 - GET THE RE-START FILE
9     Please select from the menu :
10   6
11        6
12
```

```
13    *** WHAT SHALL WE DO NOW ?
14        0 - QUIT
15        1 - PLOT THE MESH
16        2 - SMOOTH THE MESH
17        3 - SWAP DIAGONALS
18        4 - EAT 3 S
19        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
20        6 - GET THE RE-START FILE
21     Please select from the menu :
22   0
23        0
24
25   Stop
```

Should the user want to reset or obtain a new adaptive mesh, press "y" or "Y" to start over again and "n" or "N" to continue the calculation using the adapted mesh.

```
1   Do you want to resize the element again? (Y/N)
2   N
```

The hiflow program will then use the new .DAT file which contain the adapted mesh and problem information for calculation. The calculation procedure in the hiflow is the same for any n^{th} version of the mesh. The density fringe, velocity vector, pressure, and residual plot of the calculation based on 1st adaptive mesh are shown in the figure 30, 31, 32, and 33.
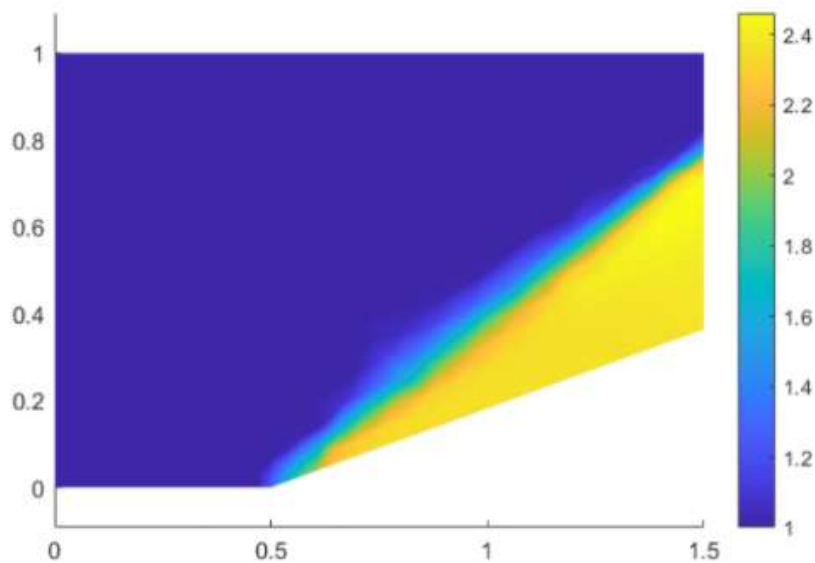

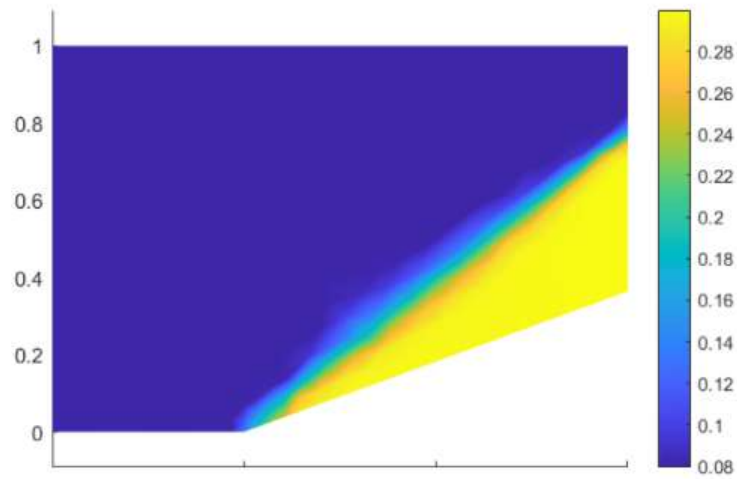Figure 30: Flow over Wedge Problems 1st Adaptive Mesh Density Plot

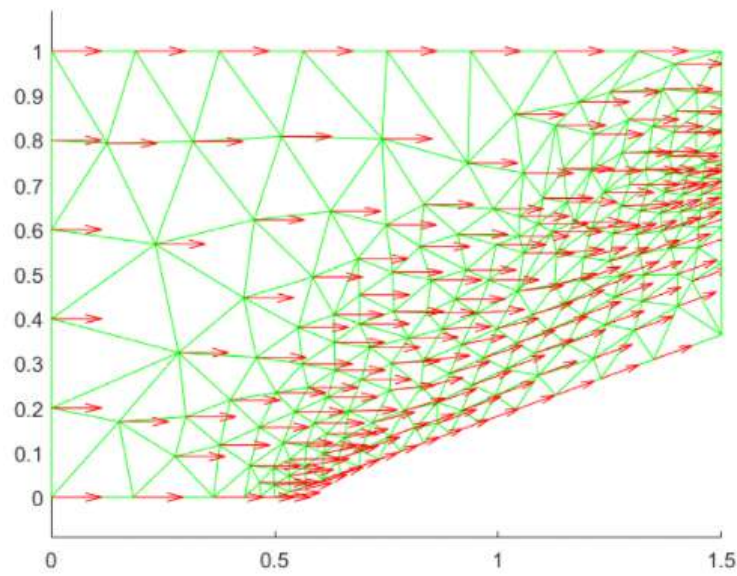Figure 31: Flow over Wedge Problems 1st Adaptive Mesh Pressure Plot

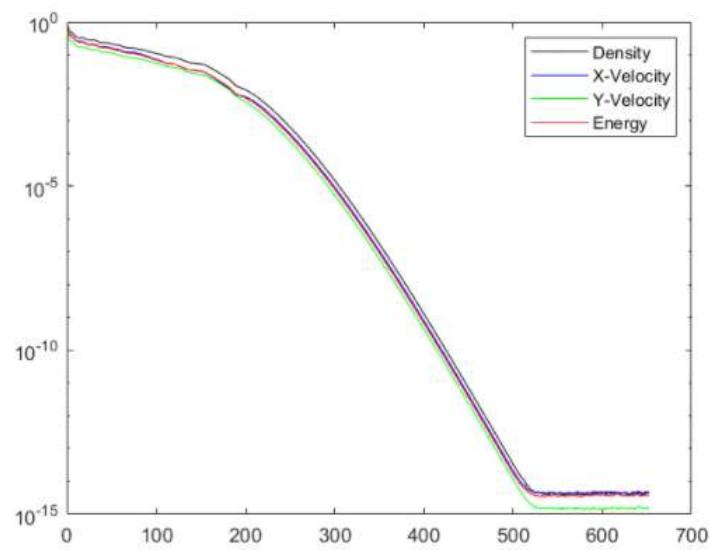Figure 32: Flow over Wedge Problems 1st Adaptive Mesh Velocity Plot

Figure 33: Flow over Wedge Problems 1st Adaptive Mesh Residual Plot

Now, applying another adaptive remesh based on the solution obtain from 1st.

```
1   *** READING DATA
2   wedg.RE3
3        0.1400      0.0100      0.0200
4
5   Input maximum element size
6   0.25
7   Input minimum element size
8   0.02
9   Input checking element size
10  0.02
11   ENTER MAXIMUM STRETCHING
12        3
13
14  Input cell aspect ratio
15  3
16   WHAT DO YOU WANT TO USE AS KEY VARIABLE ?
17   1 - DENSITY
18   2 - VELOCITY (MODULUS)
19   3 - DENSITY + VELOCITY
20   4 - ENTROPY
21   5 - DENSITY + MACH NO
22   6 - MACH NUMBER
23  Please input the key variable for adaptive remeshing :5
24        5
25
26   enter the value of gamma (usually 1.4)1.4
27  How many smoothing loops?
28  How many smoothing loops?
29   DO YOU WANT EXTRA REFINEMENT FOR STAGNATION POINTS ?
30        4
31
32  ENTER DELKM, THRESHOLD MACH NUMBER AND GAMMA
33        3    2     3
34
35   *** FILLING IN THE GAPS
36   *** FILLING ISIDE FOR THE BACKGROUND GRID
37   *** FILLING ELEMENT CONECTIVITY MATRIX
38  Input expansion factor (generally close to 0) :
39  0.1
40  AE =
41      0.0317
42
43   *** GENERATING BOUNDARY NODES
44  Current progress in interpolation 20
45  Current progress in interpolation 40
46  Current progress in interpolation 6.000000e+01
47  Current progress in interpolation 80
48  Current progress in interpolation 100
49   *** INTERPOLATING FROM THE BACKGROUND GRID
```

```
50  aw for re-ordering, (aw=1.:always,aw=large:never)
51      3
52
53   *** GENERATING BOUNDARY NODES
54   *** WHAT SHALL WE DO NOW ?
55      0 - QUIT
56      1 - PLOT THE MESH
57      2 - SMOOTH THE MESH
58      3 - SWAP DIAGONALS
59      4 - EAT 3 S
60      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
61      6 - GET THE RE-START FILE
62   Please select from the menu :
63  2
64      2
65
66  ENTER NUMBER OF SMOOTHING LOOPS
67  3
68   *** WHAT SHALL WE DO NOW ?
69      0 - QUIT
70      1 - PLOT THE MESH
71      2 - SMOOTH THE MESH
72      3 - SWAP DIAGONALS
73      4 - EAT 3 S
74      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
75      6 - GET THE RE-START FILE
76   Please select from the menu :
77  3
78      3
79
80  20.000000 sides have been swapped
81  2.000000 sides have been swapped
82  2.000000 sides have been swapped
83  0.000000 sides have been swapped
84  filling iside
85   *** WHAT SHALL WE DO NOW ?
86      0 - QUIT
87      1 - PLOT THE MESH
88      2 - SMOOTH THE MESH
89      3 - SWAP DIAGONALS
90      4 - EAT 3 S
91      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
92      6 - GET THE RE-START FILE
93   Please select from the menu :
94  4
95      4
96
97  nr. of 3's removed = 0 *** WHAT SHALL WE DO NOW ?
98      0 - QUIT
99      1 - PLOT THE MESH
```

```
100        2 - SMOOTH THE MESH
101        3 - SWAP DIAGONALS
102        4 - EAT 3 S
103        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
104        6 - GET THE RE-START FILE
105     Please select from the menu :
106    5
107        5
108
109    The checking was successful
110    Total number of generated points : 198.000000
111    Total number of generated elements : 346.000000
112     *** WHAT SHALL WE DO NOW ?
113        0 - QUIT
114        1 - PLOT THE MESH
115        2 - SMOOTH THE MESH
116        3 - SWAP DIAGONALS
117        4 - EAT 3 S
118        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
119        6 - GET THE RE-START FILE
120     Please select from the menu :
121    6
122        6
123
124     *** WHAT SHALL WE DO NOW ?
125        0 - QUIT
126        1 - PLOT THE MESH
127        2 - SMOOTH THE MESH
128        3 - SWAP DIAGONALS
129        4 - EAT 3 S
130        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
131        6 - GET THE RE-START FILE
132     Please select from the menu :
133    0
134        0
135
136    stop
137    Do you want to resize the element again? (Y/N)
138    N
```

User can then reapply adaptive remesh based on previous solution again to obtain an even more optimized mesh as shown in figure 34. For each adaptive mesh generated, the user can use .DAT (version) file to start calculate immediately without having to start from initial mesh again. Should the program encounter a problem during the calculation, it will prompt the user to select whether to reduce the relaxation factors (variable name csafe) or stop the calculation and used another input file. User can input r to reduce the relaxation factor. The initial value of this is 0.5 and can be reduced until it reaches zero. At that point, the hiflow program will stop solving for the solution iteratively and print the solution obtained during the first loop iteration. As the program will not use the old unknown variables to calculate the next iteration anymore. If the reduced csafe reaches 0.1, it is recommended that the user improve the mesh quality by redoing the adaptive remesh process again.
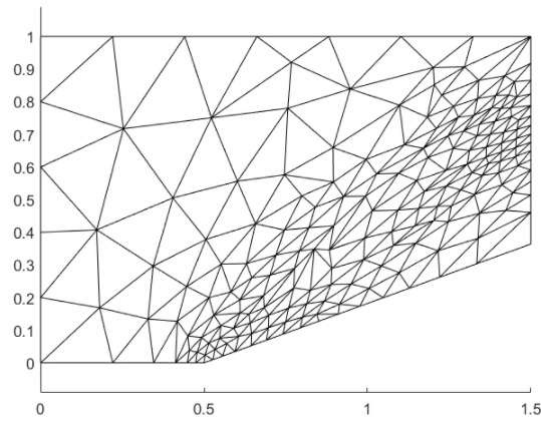
35

Figure 34: Flow over Wedge Problems 2nd Adaptive Mesh Plot

```
Error detected in hiflow, (r)educe relaxation factor or (s)top the
calculation
r
csafe =

    0.4500
```

If the reduced `csafe` is not enough, the user will be asked to reduce the relaxation factor again or stop the calculation entirely. The user should then generate the new initial mesh or change the adaptive mesh parameter.

```
Error detected in hiflow, (r)educe relaxation factor or (s)top the
calculation
s
Stopping the calculation, please use new input data>>
```

The process can be repeated as many as the user desired. Once the mesh cannot be any finer, the final mesh plot can be seen in figure 35.
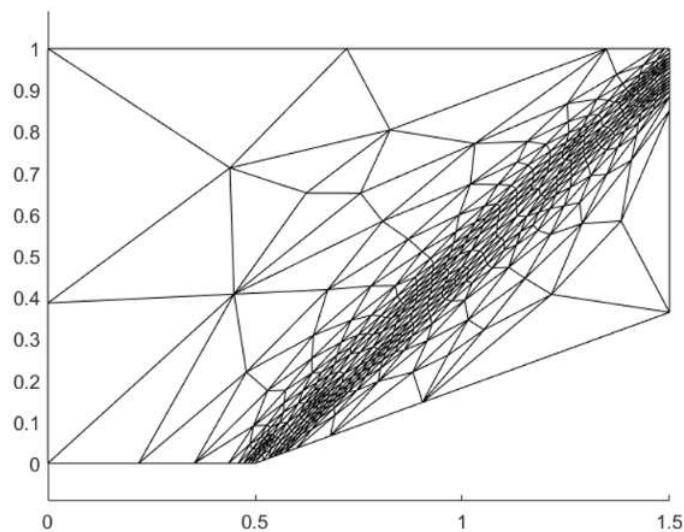

Figure 35: Flow over Wedge Problems Final Adaptive Mesh Plot

4.3.2 Flow over cylinder

The flow over cylinder problem is similar to flow over wedge problem. In contrast, this problem has more complexity since, in flow over wedge problem, the angle of the wedge slope has more value than the maximum angle that can generate shock wave line from the slope. Then, the flow over cylinder will create a shock wave curve at the front of the cylinder where, inside the area between the shock wave curve and the cylinder, the fluid velocity will become subsonic. Next, in the area after this area, the fluid velocity will gradually increases to supersonic again.

From figure 36, the flow over cylinder problem possess an incoming flow speed of $M_1 = 6$, cylinder with radius of 1.5 m, and flow domain with the dimension of 2.7 m x 4.7 m
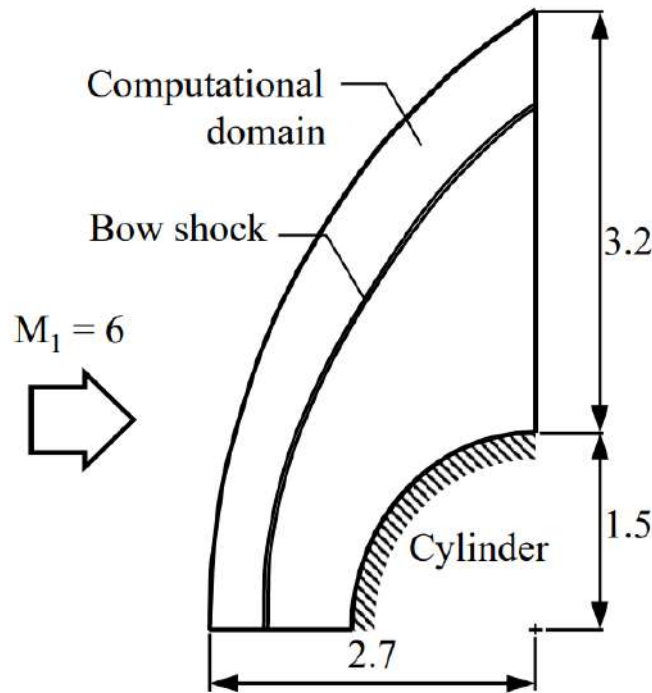

Figure 36: Flow over Cylinder Problem Statement

4.3.2.1 Input files generation Similar to 4.3.1, in this example we will focus on the solution to the flow over cylinder problem. The overall process follow the same principle as shown in the flow over wedge problem. However, to accurately create a flow domain surrounding a curve surface, a series of points arranged in a small interval is needed.

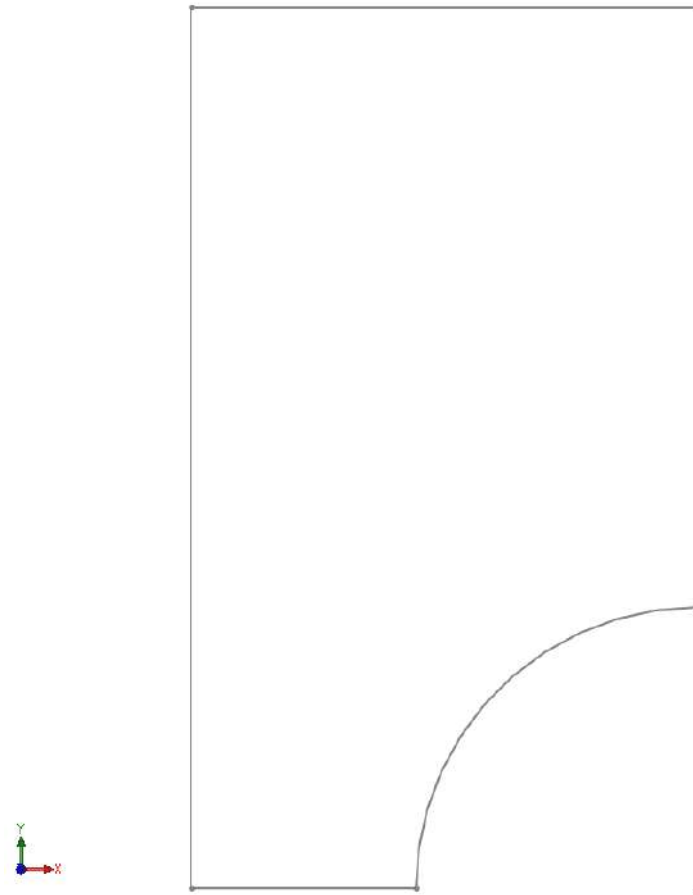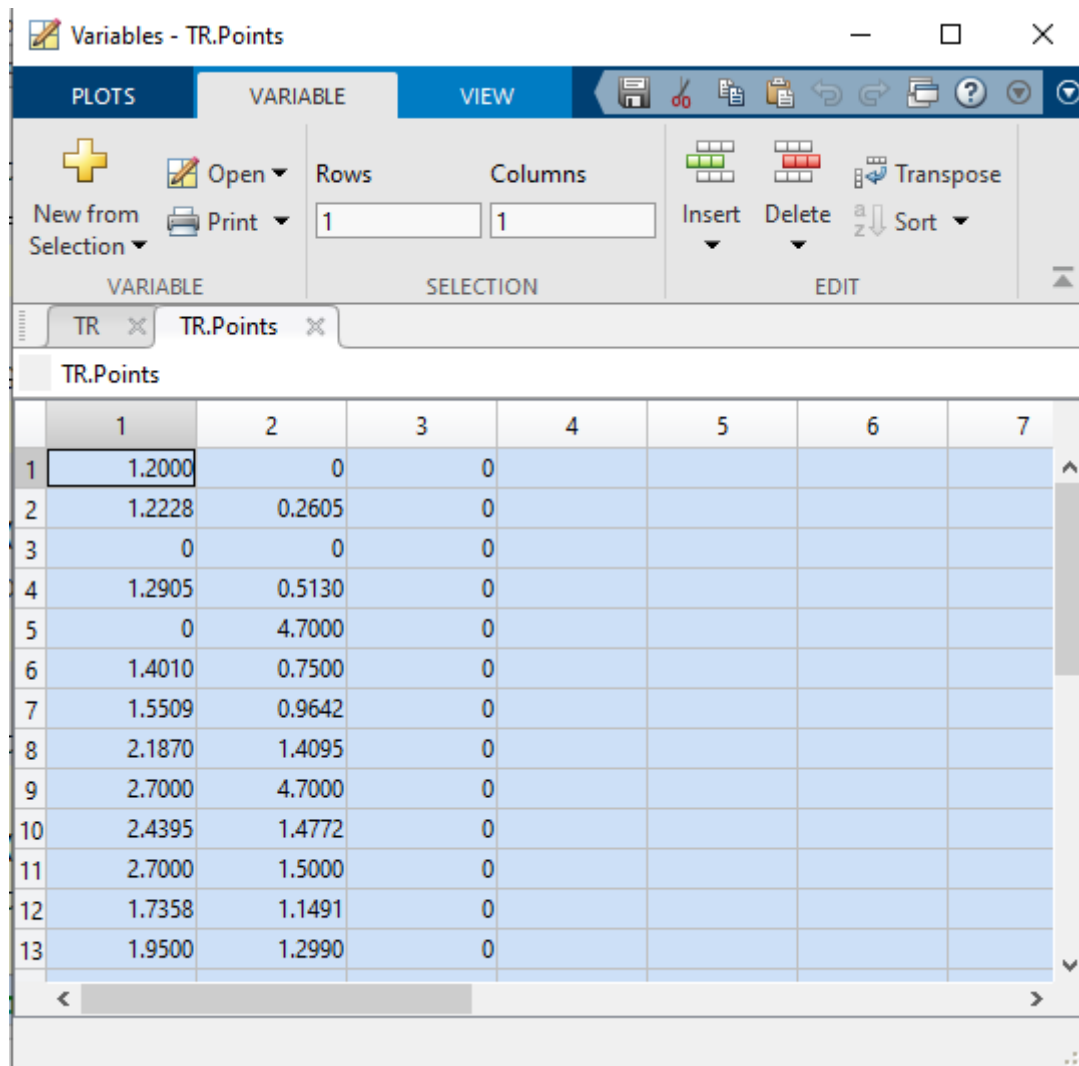From the figure 37, the user can draw the shape of the problem in the CAD [12] of choice.

Figure 37: The CAD of the Flow over cylinder body

Then, the user have to export the file in .STL format which will be used as an input in the MATLAB script shown below.

```matlab
%% STL input and visualization
% Create pde Model
model = createpde;

% Read STL file
importGeometry(model,'cy.STL');

% Plot STL file
figure(1)
pdegplot(model,'VertexLabels','on','EdgeLabels','on','FaceLabels','on')

% Read STL file
TR = stlread('cy.STL');
```

After the user run the script, the variable TR can be inspect in MATLAB Workspace as shown in the figure 38. The variable TR.points contains all points generated from .STL file.

	1	2	3	4	5	6	7
1	1.2000	0	0				
2	1.2228	0.2605	0				
3	0	0	0				
4	1.2905	0.5130	0				
5	0	4.7000	0				
6	1.4010	0.7500	0				
7	1.5509	0.9642	0				
8	2.1870	1.4095	0				
9	2.7000	4.7000	0				
10	2.4395	1.4772	0				
11	2.7000	1.5000	0				
12	1.7358	1.1491	0				
13	1.9500	1.2990	0				

Figure 38: Output variable TR

Since the variable TR.points did not come in the correct order, the user have to manually rearrange the coordinate. Figure 39 shows the rearranged coordinates of .FIX file of this problem.

```
1    NREG    NFN    NBCS
2    1         13     13
3    COORDINATES
4    1 0.0                      0.0
5    2 1.20000004768372         0.0
6    3 1.22278833389282         0.260472267866135
7    4 1.29046106338501         0.513030230998993
8    5 1.400961875911553        0.750000000000000
9    6 1.55093336105347         0.964181423187256
10   7 1.73581862449646         1.149066668663025
11   8 1.95000004768372         1.29903805255890
12   9 2.18696975708008         1.40953898429871
13   10 2.43952775001526        1.47721159458160
14   11 2.70000004768372        1.50000000000000
15   12 2.70000004768372        4.69999980926514
16   13 0.0                      4.69999980926514
```

Figure 39: Coordinate of the Flow over cylinder body

For .RE file, an alternative method is used here to generate the initial triangular element. Instead of create a small element that connect every initial vertices, we will instead create a two large triangular element that completely overlapped the flow domain. Figure 40 shows a method which mapped the mentioned triangular element over the problem domain.

The key to this method is to generate a mesh large enough to fully encompass the entire flow domain of the problem. The points used to define these elements can be added and do not need to be the same as those defined in the .FIX file. Figure 41 shows .RE file using this method.

Again, the process followed the same principle as discussed earlier. Even with an alternative method, the information to be input in the file is the new element based on the method that the user uses. Note that the parameter N2BODY for this method is set to a default value of 2. This is an artifact from the original code flow that should be examined more in the future.
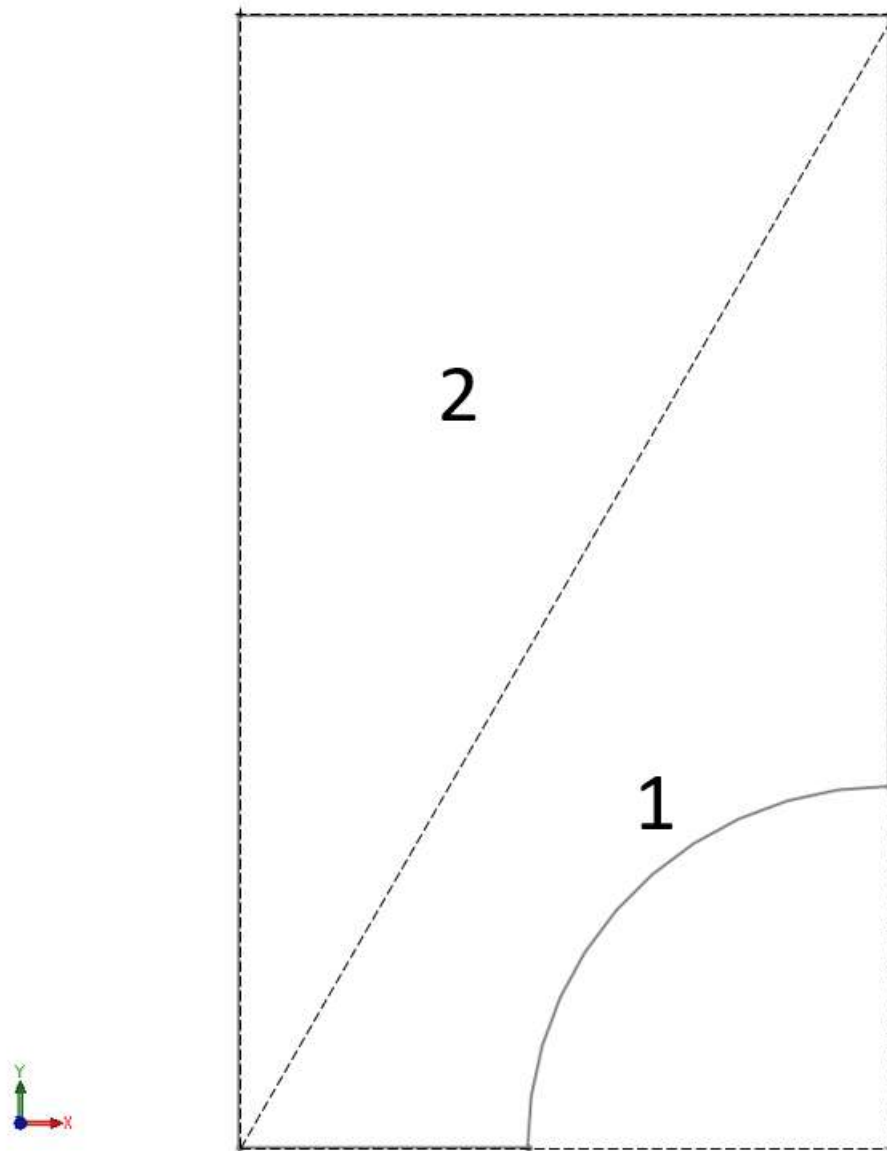
Figure 40: Alternative initial meshing method

Figure 41: Example .RE file using alternative method for flow over cylinder problem

41

We will first begin by running the `hiflow` program and selecting associate `cylinder` files.

```
1   .(d)at or .(f)ix data input?: f
2   Enter the input geometry file name: cylinder
3   ----------------------------------------------------
4    *** READING DATA
5   cylinder.RE1
6   Input initial uniform mesh size
7   0.1
8    *** FILLING IN THE GAPS
9    *** FILLING ISIDE FOR THE BACKGROUND GRID
10   *** FILLING ELEMENT CONECTIVITY MATRIX
11  Input expansion factor (generally close to 0) :
12  0.1
13  AE =
14      2.7000
15
16   *** GENERATING BOUNDARY NODES
17   *** INTERPOLATING FROM THE BACKGROUND GRID
18  aw for re-ordering, (aw=1.:always,aw=large:never)
19      2
20
21   *** GENERATING BOUNDARY NODES
22   *** WHAT SHALL WE DO NOW ?
23      0 - QUIT
24      1 - PLOT THE MESH
25      2 - SMOOTH THE MESH
26      3 - SWAP DIAGONALS
27      4 - EAT 3 S
28      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
29      6 - GET THE RE-START FILE
30   Please select from the menu :
31  2
32      2
33
34  ENTER NUMBER OF SMOOTHING LOOPS
35  30
36   *** WHAT SHALL WE DO NOW ?
37      0 - QUIT
38      1 - PLOT THE MESH
39      2 - SMOOTH THE MESH
40      3 - SWAP DIAGONALS
41      4 - EAT 3 S
42      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
43      6 - GET THE RE-START FILE
44   Please select from the menu :
45  3
46      3
47
48  73.000000 sides have been swapped
49  2.000000 sides have been swapped
```

```
50   0.000000 sides have been swapped
51   filling iside
52    *** WHAT SHALL WE DO NOW ?
53        0 - QUIT
54        1 - PLOT THE MESH
55        2 - SMOOTH THE MESH
56        3 - SWAP DIAGONALS
57        4 - EAT 3 S
58        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
59        6 - GET THE RE-START FILE
60    Please select from the menu :
61   4
62        4
63
64   nr. of 3's removed = 0
65    *** WHAT SHALL WE DO NOW ?
66        0 - QUIT
67        1 - PLOT THE MESH
68        2 - SMOOTH THE MESH
69        3 - SWAP DIAGONALS
70        4 - EAT 3 S
71        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
72        6 - GET THE RE-START FILE
73    Please select from the menu :
74   5
75        5
76
77   The checking was successful
78   Total number of generated points : 1229.000000
79   Total number of generated elements : 2308.000000
80    *** WHAT SHALL WE DO NOW ?
81        0 - QUIT
82        1 - PLOT THE MESH
83        2 - SMOOTH THE MESH
84        3 - SWAP DIAGONALS
85        4 - EAT 3 S
86        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
87        6 - GET THE RE-START FILE
88    Please select from the menu :
89   6
90        6
91
92    *** WHAT SHALL WE DO NOW ?
93        0 - QUIT
94        1 - PLOT THE MESH
95        2 - SMOOTH THE MESH
96        3 - SWAP DIAGONALS
97        4 - EAT 3 S
98        5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
99        6 - GET THE RE-START FILE
```

```
100    Please select from the menu :
101   0
102        0
103
104   stop
105   Continue:? (Y/N) y
```

Figure 42 shows the generated mesh for the flow over cylinder problem using the initial uniform mesh size of 0.1 and expansion factor of 0.1. The generated mesh is now ready for the calculation.
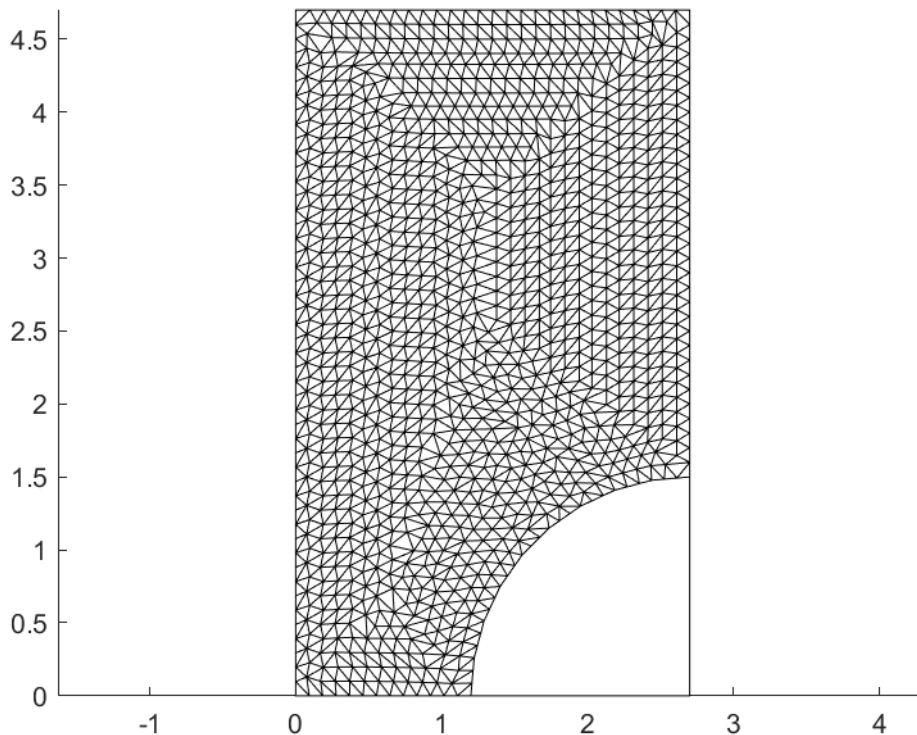

Figure 42: Flow Over Cylinder Problem Uniform Mesh Plot

Next, the hiflow program will start calculating based on the problem definition and the mesh obtained earlier. A figure showing the current residual plot will pop-up automatically for users to observe the real time updating of calculation residual as shown in figure 43. If the current residual values are satisfactory enough, simply press "f" to stop the calculation immediately.

```
1    The finite element model consists of:
2        number of nodes                 =     1229
3        number of elements              =     2308
4        number of boundary conditions   =      148
5        number of iterations needed     =    10000
6
7     Performing iterations for convergence
8      Iter      Del rho     Del rho-u    Del rho-v     Del rho-e
9         1 2.78852e+00 9.88130e-01 1.53687e+00 1.52426e+00
10       10 1.51041e+00 4.38045e-01 6.73744e-01 8.14118e-01
11       20 1.48745e+00 6.00022e-01 6.34778e-01 7.80221e-01
12       30 1.27262e+00 5.67035e-01 6.28720e-01 6.57355e-01
13       40 1.35862e+00 6.33473e-01 6.50633e-01 7.16294e-01
14                               .
```

44

```
15                                    .
16                                    .
17
18   continueCalculation =
19
20     logical
21
22       0
23
24   Do you want to continue calculating? (Y/N) :
25   y
26   How many more iteration?
27   5000
28
29   niter =
30
31         15000
32
33
34   continueCalculation =
35
36     logical
37
38       1
39                                    .
40                                    .
41                                    .
42       14960 1.85134e-14 1.75430e-14 5.78103e-15 1.08632e-14
43       14970 1.80874e-14 1.72872e-14 5.63768e-15 1.05143e-14
44       14980 1.82695e-14 1.73860e-14 5.65950e-15 1.05395e-14
45       14990 1.82786e-14 1.78815e-14 5.94100e-15 1.09479e-14
46       15000 1.84634e-14 1.77315e-14 5.46767e-15 1.06853e-14
47
48   continueCalculation =
49
50     logical
51
52       0
53
54   Do you want to continue calculating? (Y/N) :
```

It can be seen that, from code snippet above, the program asked the user whether he/she wants to continue the calculation or not. In this case, since, at that moment, the solution has not converged yet, so more iterations are needed. Figure 43 shows the iteration plot needed to achieve convergence of 10^{-15} resulting from 5,000 more iterations.

Figure 44, 45, and 46 shows the result density, velocity vector, and pressure plot of flow over cylinder problem, respectively.

Press F to stop the iteration loop
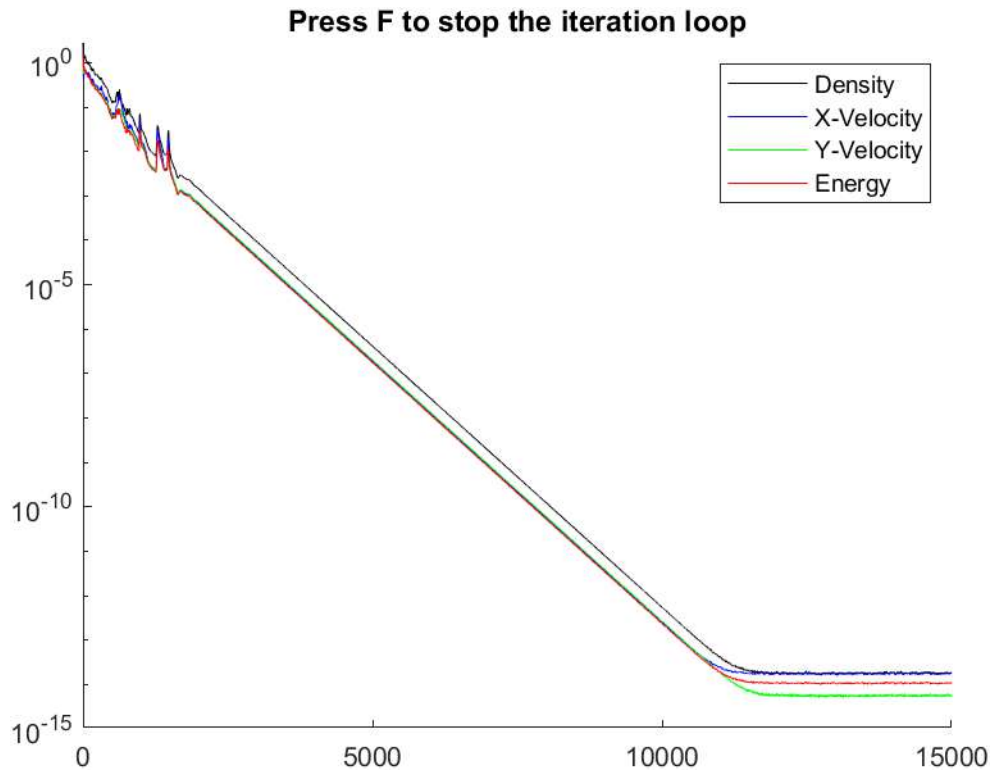
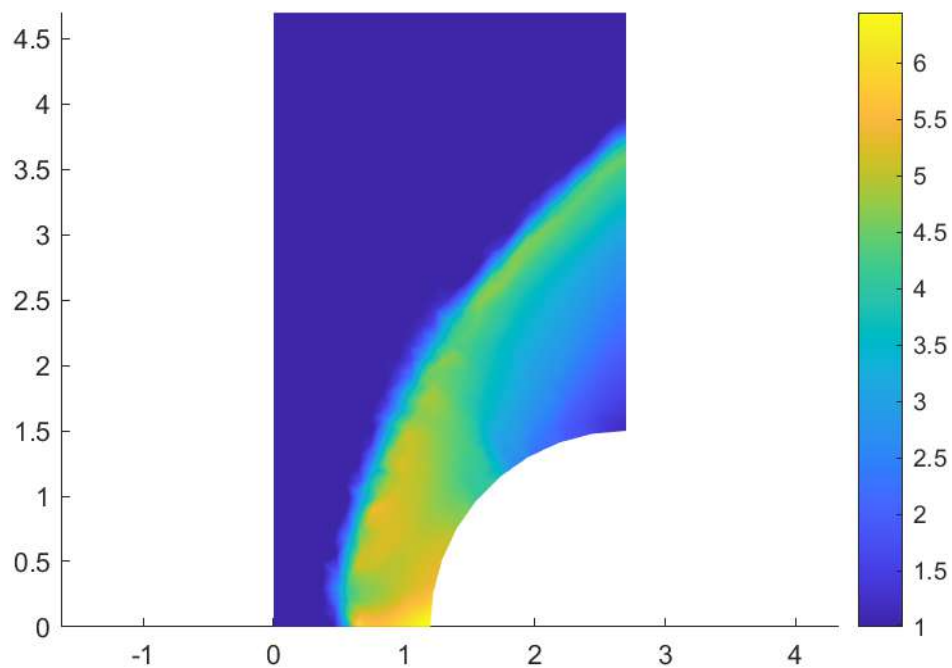Figure 43: Residual plot of Flow over Cylinder problem

Figure 44: Density plot of initial mesh for Flow over Cylinder problem

Similarly to section 4.3.1, an adaptive mesh of the problem can be generated using the solution obtained from this initial mesh.

The program will use the solution file (.RE2) obtained from the current calculation as a base for remeshing. The user must specify the maximum, minimum, checking element size, cell's maximum aspect ratio, and a key variable as a basis for the remeshing.

```
1    *** READING DATA
2  cylinder.RE2
3  Input maximum element size
```
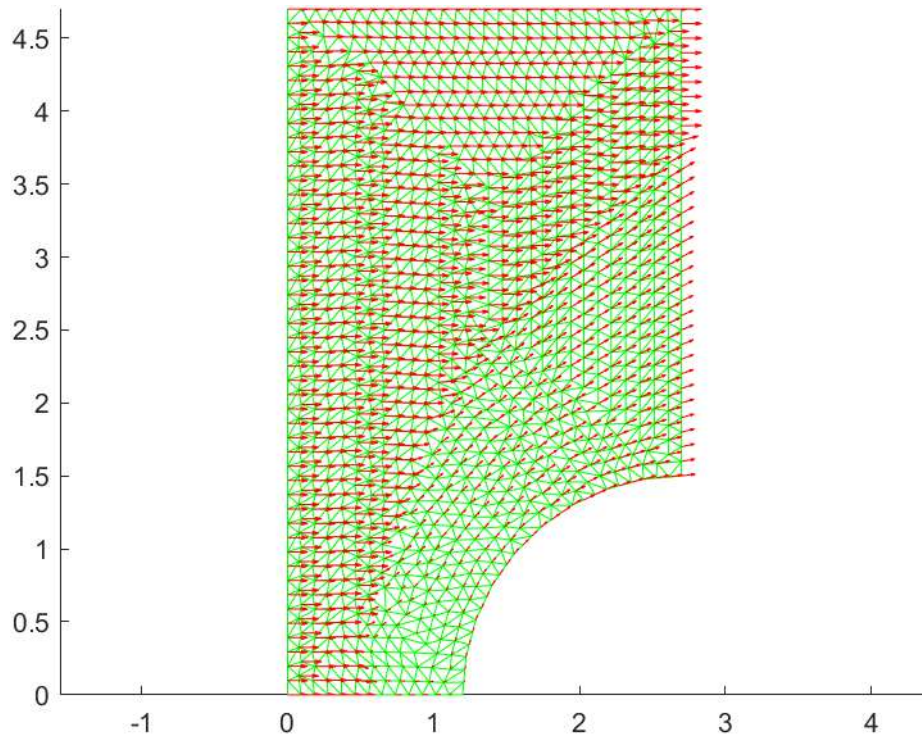
Figure 45: Velocity vector plot of initial mesh for Flow over Cylinder problem
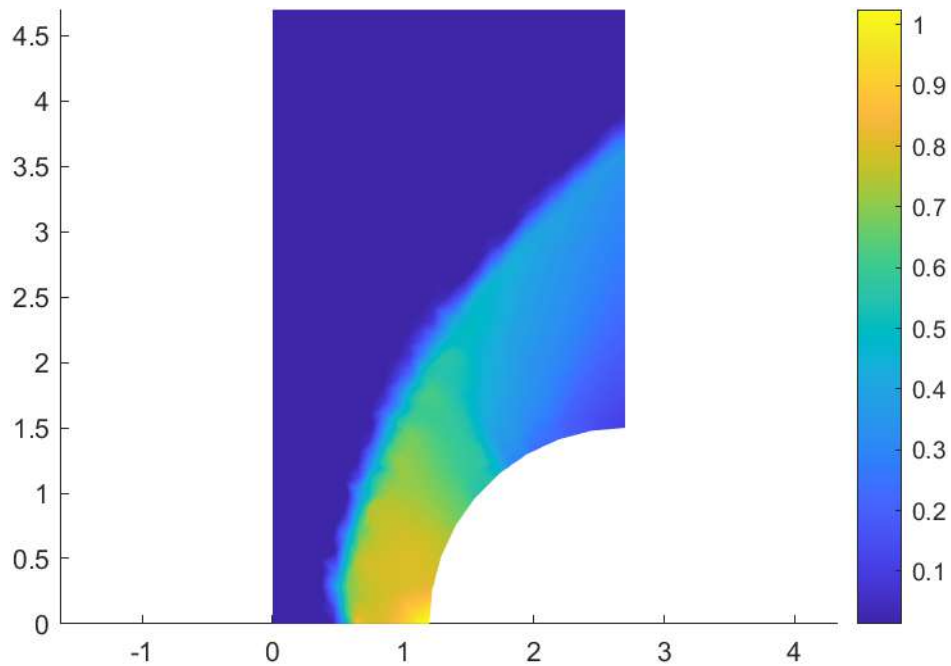
Figure 46: Pressure plot of initial mesh for Flow over Cylinder problem

```
4    0.3
5    Input minimum element size
6    0.05
7    Input checking element size
8    0.1
9     ENTER MAXIMUM STRETCHING
10   Input cell aspect ratio
11   3
12    WHAT DO YOU WANT TO USE AS KEY VARIABLE ?
```

```
1 - DENSITY
2 - VELOCITY (MODULUS)
3 - DENSITY + VELOCITY
4 - ENTROPY
5 - DENSITY + MACH NO
6 - MACH NUMBER
Please input the key variable for adaptive remeshing :5
     5

 enter the value of gamma (usually 1.4)1.4
How many smoothing loops?
How many smoothing loops?
 DO YOU WANT EXTRA REFINEMENT FOR STAGNATION POINTS ?
     4

ENTER DELKM, THRESHOLD MACH NUMBER AND GAMMA
     3     2     3

 *** FILLING IN THE GAPS
 *** FILLING ISIDE FOR THE BACKGROUND GRID
 *** FILLING ELEMENT CONECTIVITY MATRIX
Input expansion factor (generally close to 0) :
0.1
AE =
    0.0872

 *** GENERATING BOUNDARY NODES
Current progress in interpolation 7.692308e+00
Current progress in interpolation 1.538462e+01
Current progress in interpolation 2.307692e+01
Current progress in interpolation 3.076923e+01
Current progress in interpolation 3.846154e+01
Current progress in interpolation 4.615385e+01
Current progress in interpolation 5.384616e+01
Current progress in interpolation 6.153846e+01
Current progress in interpolation 6.923077e+01
Current progress in interpolation 7.692308e+01
Current progress in interpolation 8.461539e+01
Current progress in interpolation 9.230769e+01
Current progress in interpolation 100
 *** INTERPOLATING FROM THE BACKGROUND GRID
aw for re-ordering, (aw=1.:always,aw=large:never)
     3

 *** GENERATING BOUNDARY NODES
 *** WHAT SHALL WE DO NOW ?
     0 - QUIT
     1 - PLOT THE MESH
     2 - SMOOTH THE MESH
     3 - SWAP DIAGONALS
```

```
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
2
      2

ENTER NUMBER OF SMOOTHING LOOPS
10
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
3
      3

38.000000 sides have been swapped
upper
5.000000 sides have been swapped
upper
0.000000 sides have been swapped
filling iside
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
2
      2

ENTER NUMBER OF SMOOTHING LOOPS
10
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
  Please select from the menu :
```

```
3
      3

1.000000 sides have been swapped
0.000000 sides have been swapped
filling iside
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
4
      4

nr. of 3's removed = 0
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
5
      5

The checking was successful
Total number of generated points : 344.000000
Total number of generated elements : 617.000000
 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
      3 - SWAP DIAGONALS
      4 - EAT 3 S
      5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
      6 - GET THE RE-START FILE
 Please select from the menu :
6
      6

 *** WHAT SHALL WE DO NOW ?
      0 - QUIT
      1 - PLOT THE MESH
      2 - SMOOTH THE MESH
```

```
163       3 - SWAP DIAGONALS
164       4 - EAT 3 S
165       5 - AREA CHECK/OUTPUT NO OF NODES AND ELEMENTS
166       6 - GET THE RE-START FILE
167  Please select from the menu :
168  0
169       0
170
171  stop
172  Do you want to resize the element again? (Y/N)
173  n
174
175  The finite element model consists of:
176       number of nodes              =      344
177       number of elements           =      617
178       number of boundary conditions =       69
179       number of iterations needed  =    10000
180
181   Performing iterations for convergence
182    Iter       Del rho      Del rho-u      Del rho-v      Del rho-e
183      1 2.75681e+00 9.29654e-01 1.58662e+00 1.50693e+00
184     10 1.25721e+00 4.02381e-01 5.86309e-01 6.70784e-01
185     20 1.35789e+00 4.94335e-01 6.22821e-01 7.30479e-01
186     30 9.76207e-01 4.03289e-01 4.74969e-01 4.78883e-01
187      .
188      .
189      .
```

Which result in a 1st adaptive mesh with a result mesh, density, velocity vector, and pressure plot as shown in figure 47, 48, 49
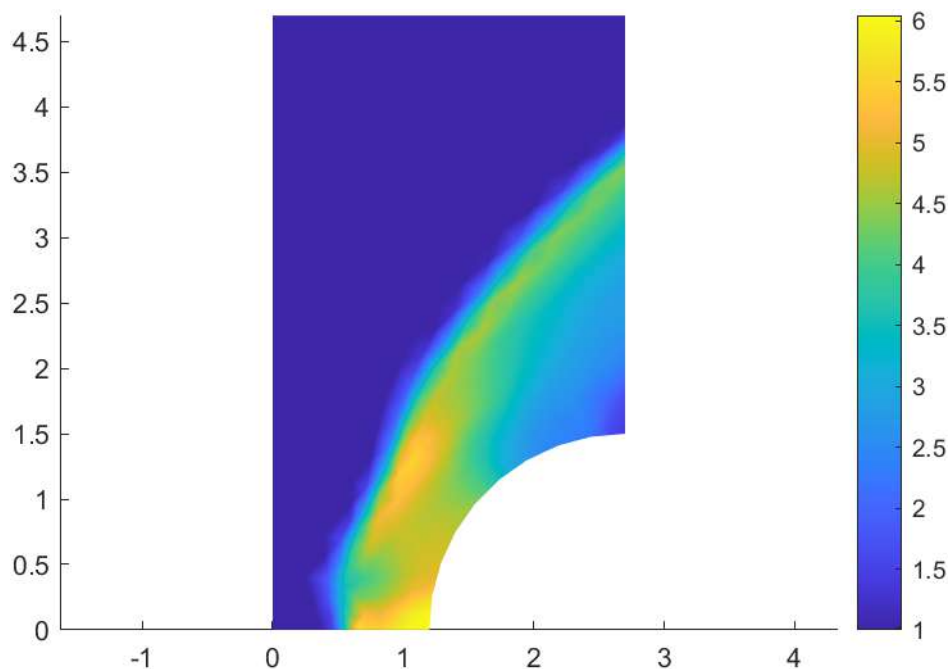
Figure 47: Density plot of 1st adaptive mesh for Flow over Cylinder problem
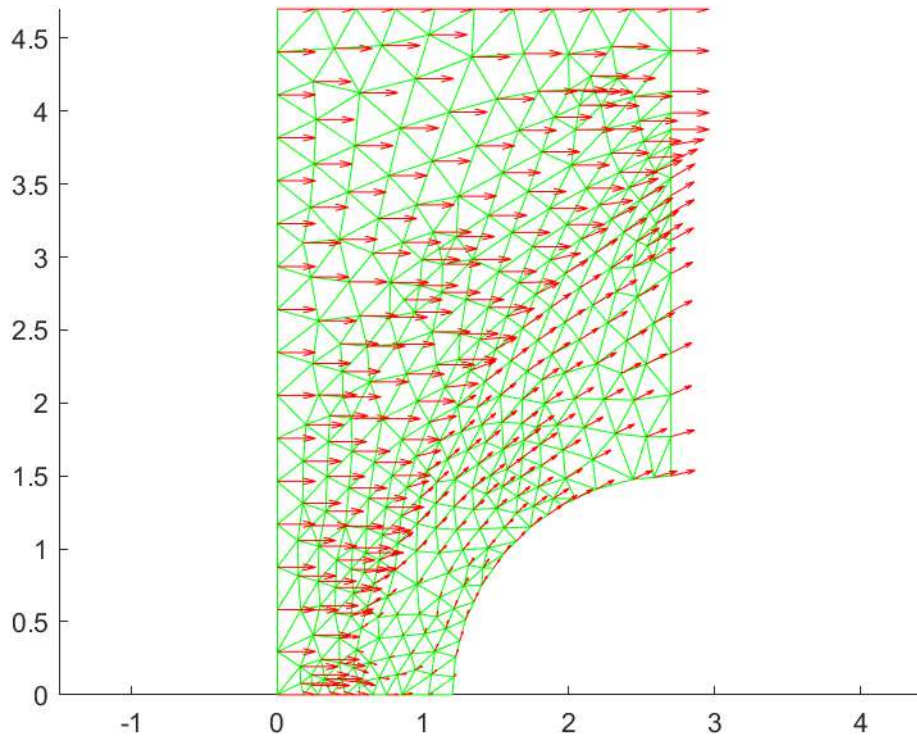
Figure 48: Velocity vector plot of 1st adaptive mesh for Flow over Cylinder problem
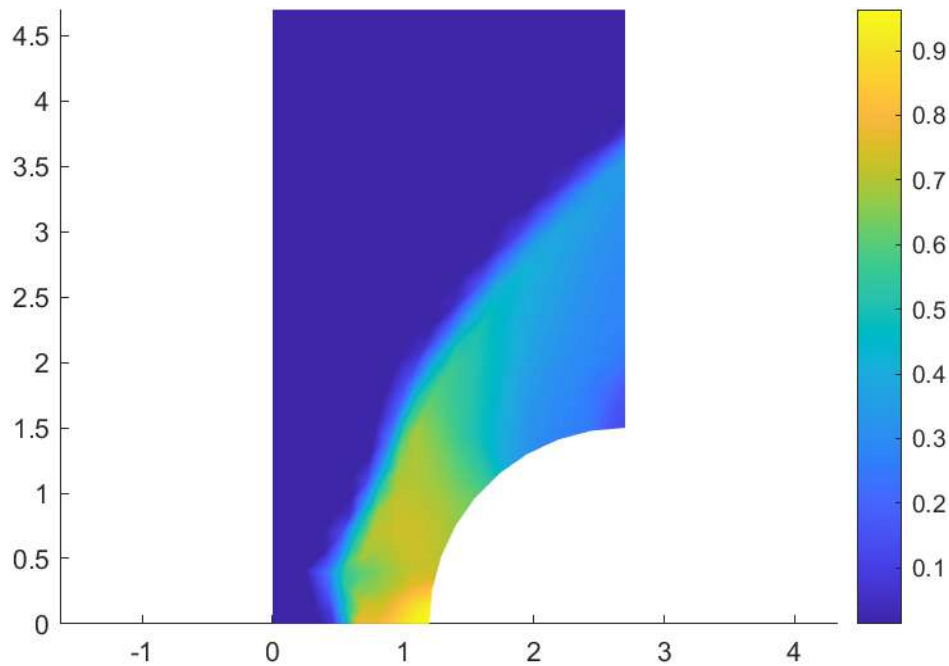
Figure 49: Pressure plot of 1st adaptive mesh for Flow over Cylinder problem

We can keep using the solution obtained from the nth adaptive mesh to calculate for the new $n + 1^{th}$ adaptive mesh. This new adaptive mesh depending on the selected adaptive variable will result in a mesh that can capture the physical phenomena more accurate. Figure 50 shows the result of using an adaptive mesh refinement method in this flow over cylinder problem. It can be seen that the accuracy of the result become progressively better as further refinement is applied.

(a) Mesh plot of 2nd adaptive mesh for Flow over Cylinder problem

(b) Density plot of 2nd adaptive mesh for Flow over Cylinder problem
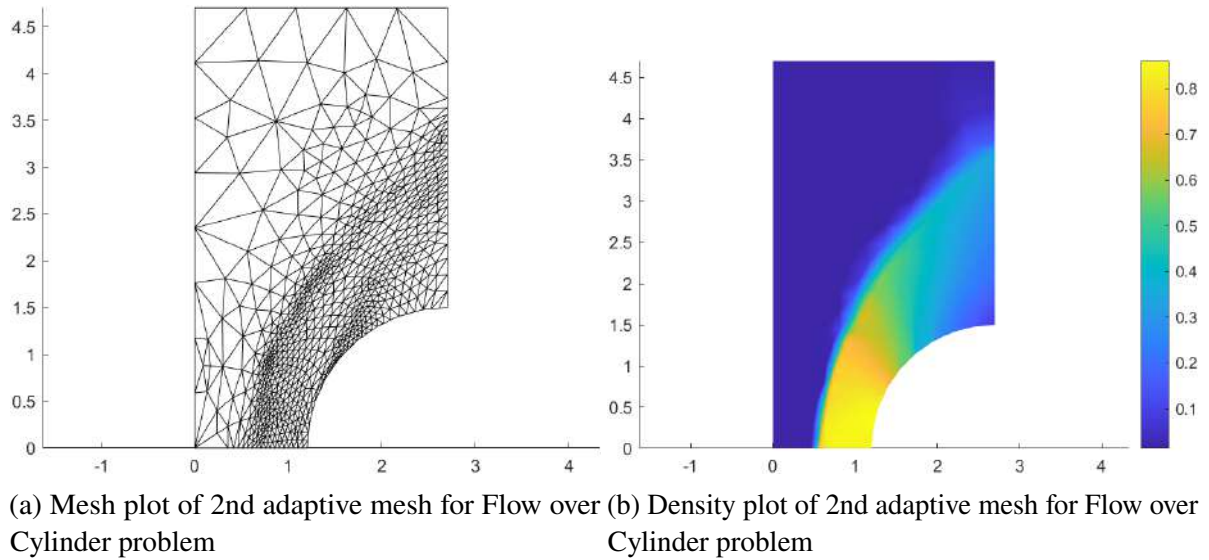
Figure 50: Point addition operation

(a) Mesh plot of 3rd adaptive mesh for Flow over Cylinder problem

(b) Density plot of 3rd adaptive mesh for Flow over Cylinder problem
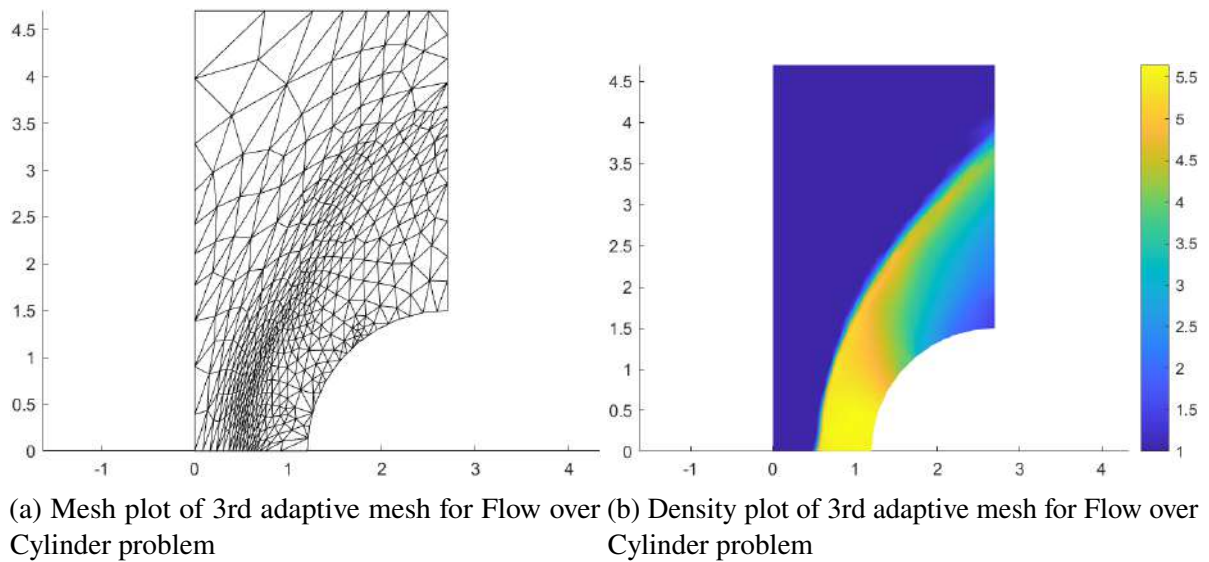
Figure 51: Point addition operation

5   Experimental Setup and Methods

5.1   Experimental Setup

To demonstrate the capabilities of the software, the developed software will be used to solve three different compressible flow problems, which are flow over a wedge, flow over a diamond-shaped airfoil, and expansion flow. The results from these three problems will be compared to the exact solution obtained from classical method to see the accuracy. The accuracy level will be determined by the flow's density values along hypothetical horizontal line from both exact and numerical solution.
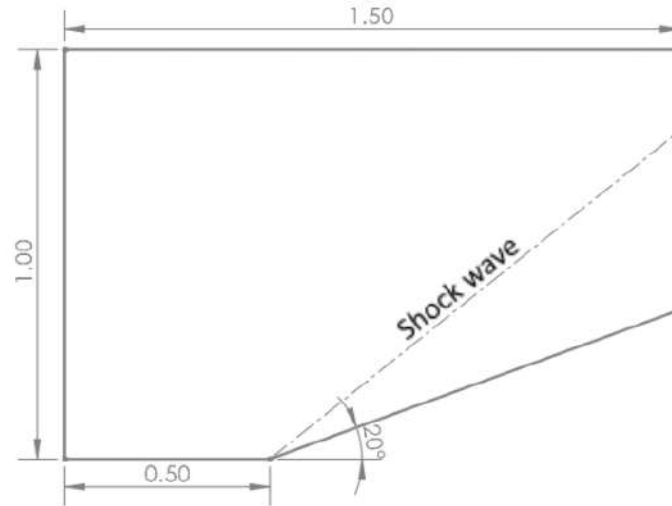
Figure 52: Flow over Wedge Problem

From figure 52, the lower width before the wedge is 0.5 m, and the upper width is 1.5 m. The height of the flow domain is 1.0 m. The wedge has an angle of 20°. Lastly, the in-flow of Mach 0.8.
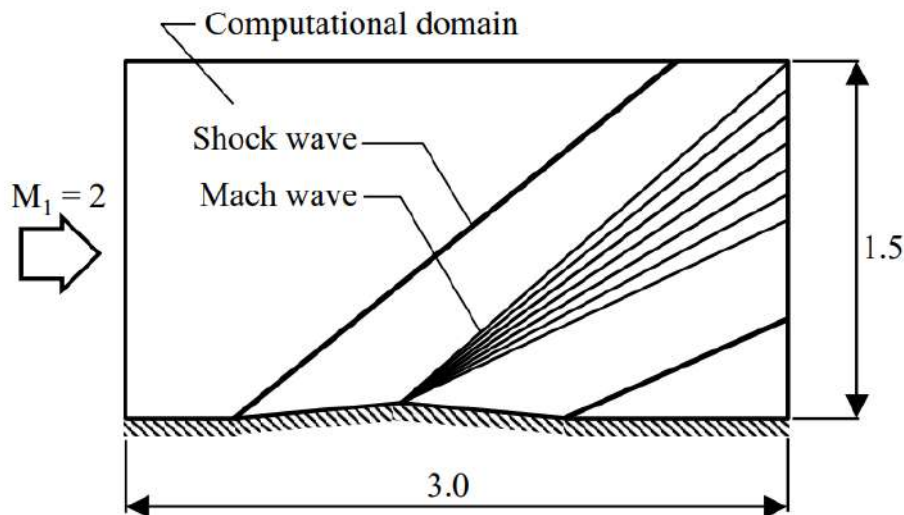
Figure 53: Flow Over a Diamond-shape Airfoil

From figure 53, the lower and upper width is 3.0 m. The height of the flow domain is 1.5 m. Lastly, the inflow of Mach 2.
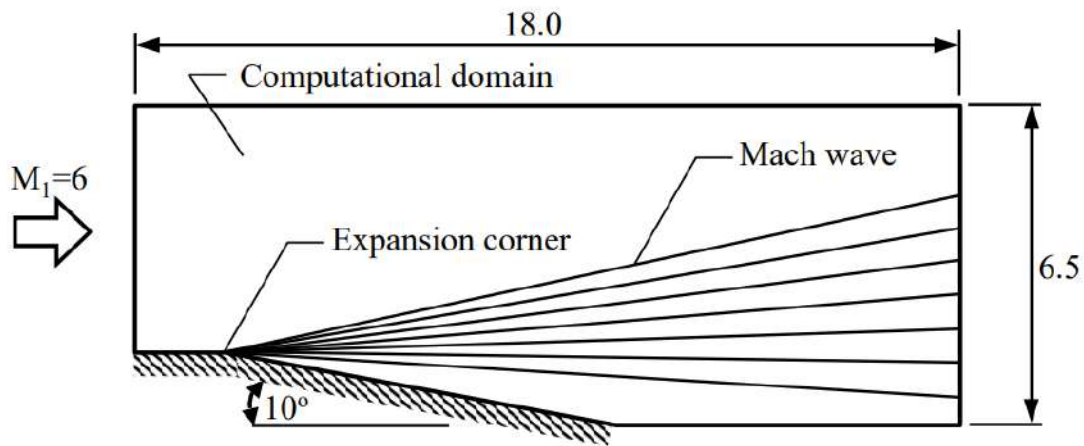
Figure 54: Expansion Flow

From figure 54,the lower and upper widths are 18.0 m. The height of the flow domain is 6.5 m. The expansion corner has an angle of 10°. Lastly, the inflow of Mach 6.

5.2 Experimental Methods

Using the problems in section 5.1, the problem is solved using the developed software. The process of calculation will be similar across these problem since the program is data-driven. The use of the software can be reviewed in Section 4.3.1. The objective of the experiment is to compare the density solution generated from the developed software with the exact solution.

6 Results and Discussion

6.1 Results

6.1.1 Flow Over Wedge

The initially generated mesh of flow over wedge problem can be seen from figure 55.
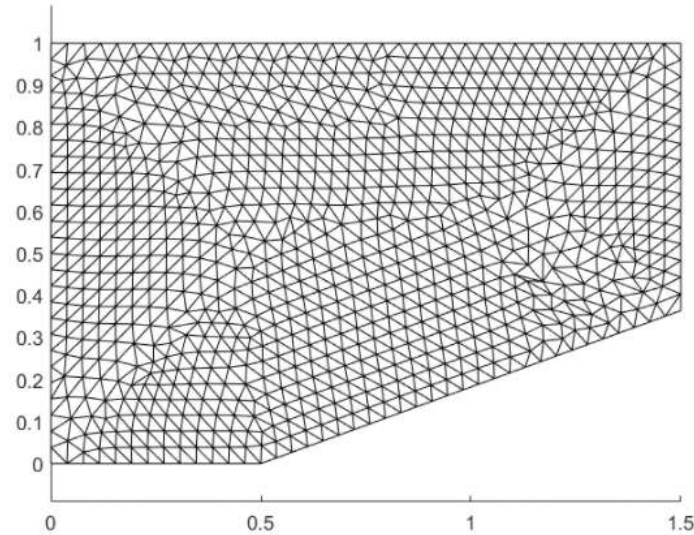
Figure 55: Flow over Wedge Problems Mesh Plot (result)

The distribution of density in the flow domain solved using the originally generated mesh can be seen in figure 56. It can be seen that the flow changed direction as it traveled along with the wedge. The change in flow's direction causes the compression of the air which creates a shock wave line forming at the start of the wedge. From the gradient of the graph, it can be observed that the value of the density in the area before the shock wave line is drastically higher than that before the shock wave line. However, it can also be seen that the area of the shock wave line expands to a great width, which is due to the element size along the shock wave line being too large.
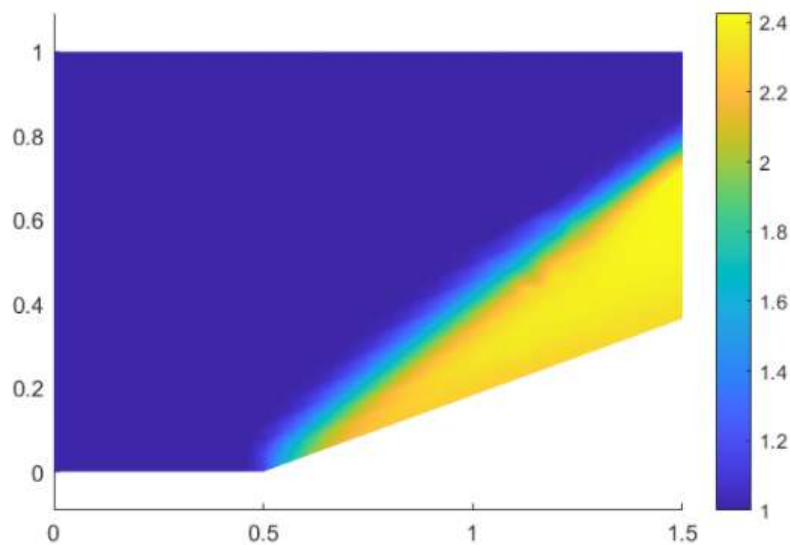
Figure 56: Flow over Wedge Problems Density Plot (result)

After applying the adaptive meshing, the 1st Adaptive Mesh of flow over wedge problems can be seen in figure 57. It can be seen that the elements inside the plot possess different sizes. Moreover, the elements residing along the shock wave line have a smaller size compared to that of the other parts.
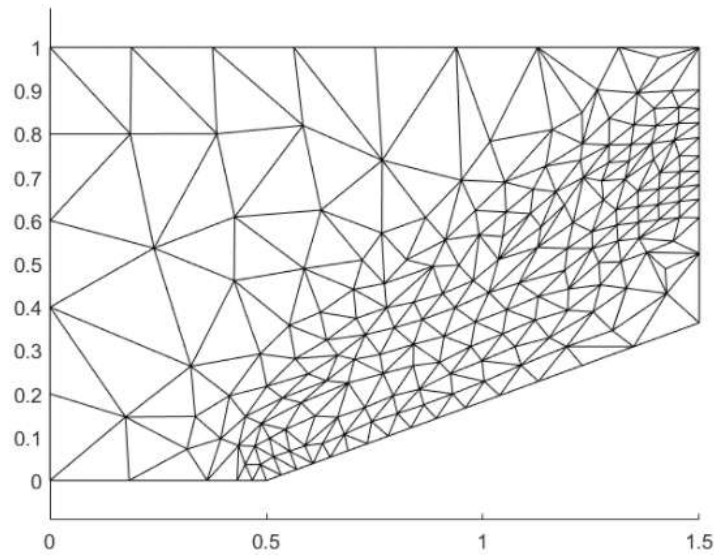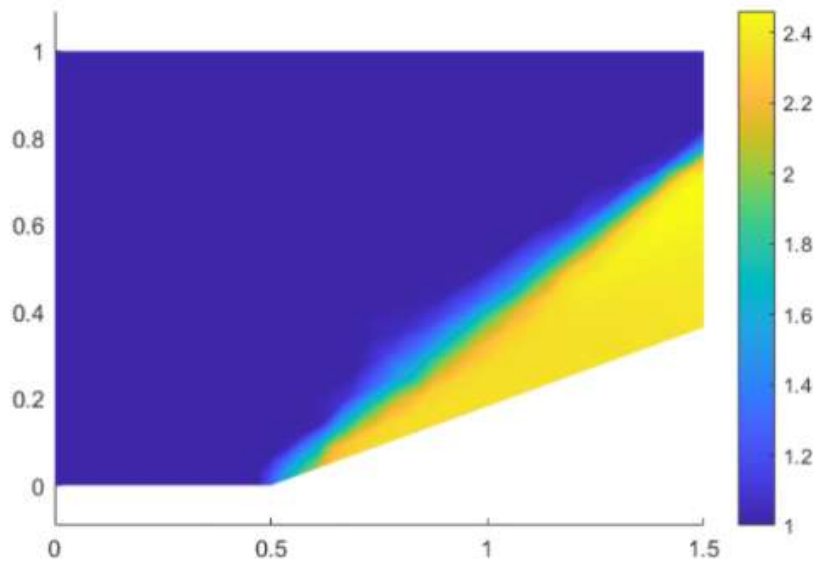

Figure 57: Flow over Wedge Problems 1st Adaptive Mesh (result)

Using the newly generated mesh, the density plot can be determined as shown in figure 58. Compared with the result from the figure 56, it can be seen that the shock wave line becomes slimmer.

Figure 58: Flow over Wedge Problems 1st Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 2nd Adaptive Mesh of flow over wedge problems can be seen in figure 59. It can be seen that the elements residing along the shock wave line possess even smaller size compared to the previous plot (figure 57).
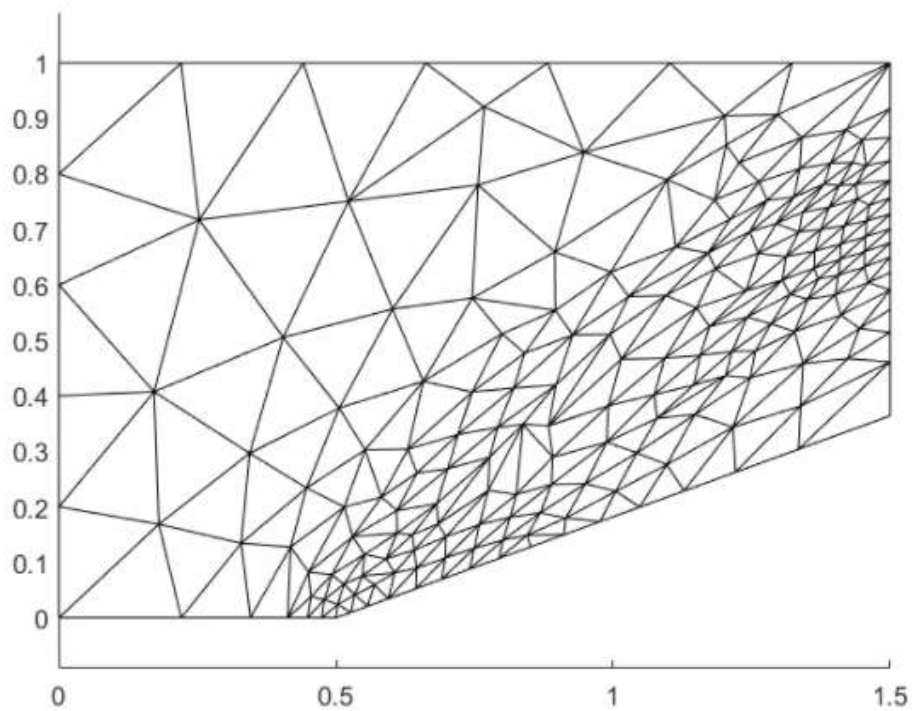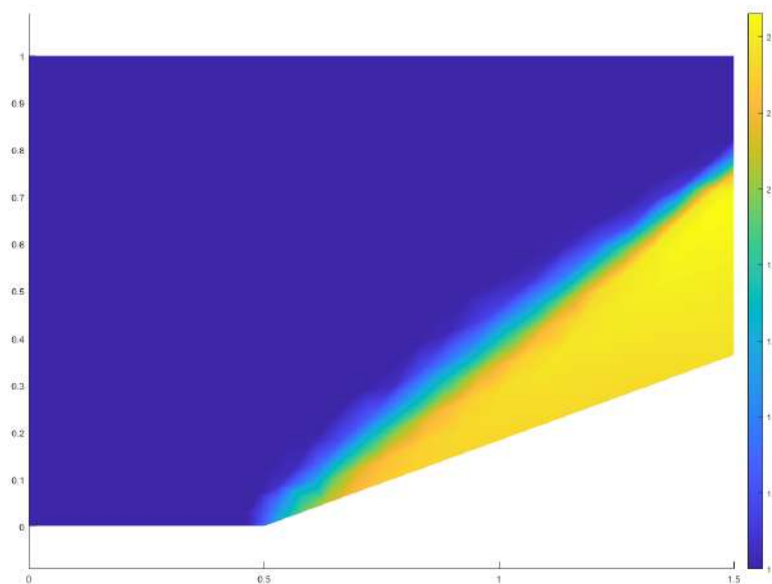

Figure 59: Flow over Wedge Problems 2nd Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in figure 60. Compared with the result from the figure 56, it can be seen that the shock wave line becomes even slimmer.

Figure 60: Flow over Wedge Problems 2nd Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 3rd Adaptive Mesh of flow over wedge problems can be seen in figure 61. It can be seen that the elements residing along the shock wave line form themselves into straight column.
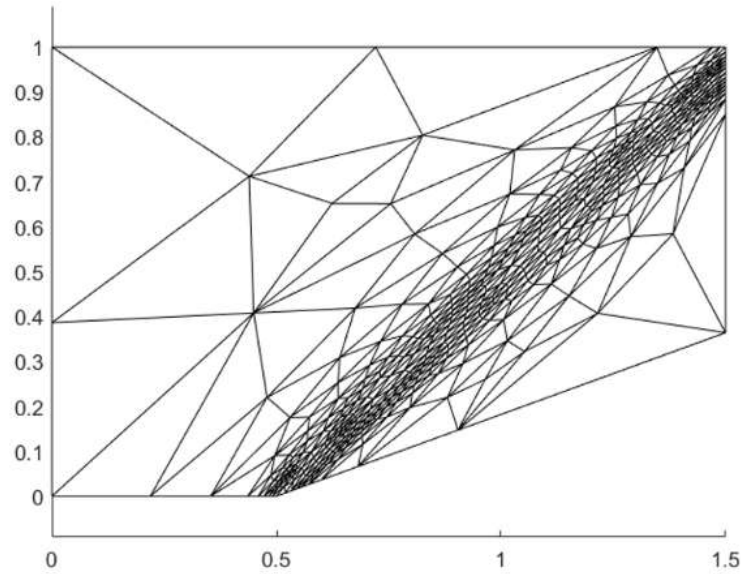

Figure 61: Flow over Wedge Problems Final Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in figure 62. Compared with the result from figure 60, it can be seen that the shock wave line becomes even slimmer than before. Moreover, the shock wave line divided the flow domain into 2 distinct parts: the area at the front of the shock wave line where the density is low and the area at the back of the shock wave line where the density is high.
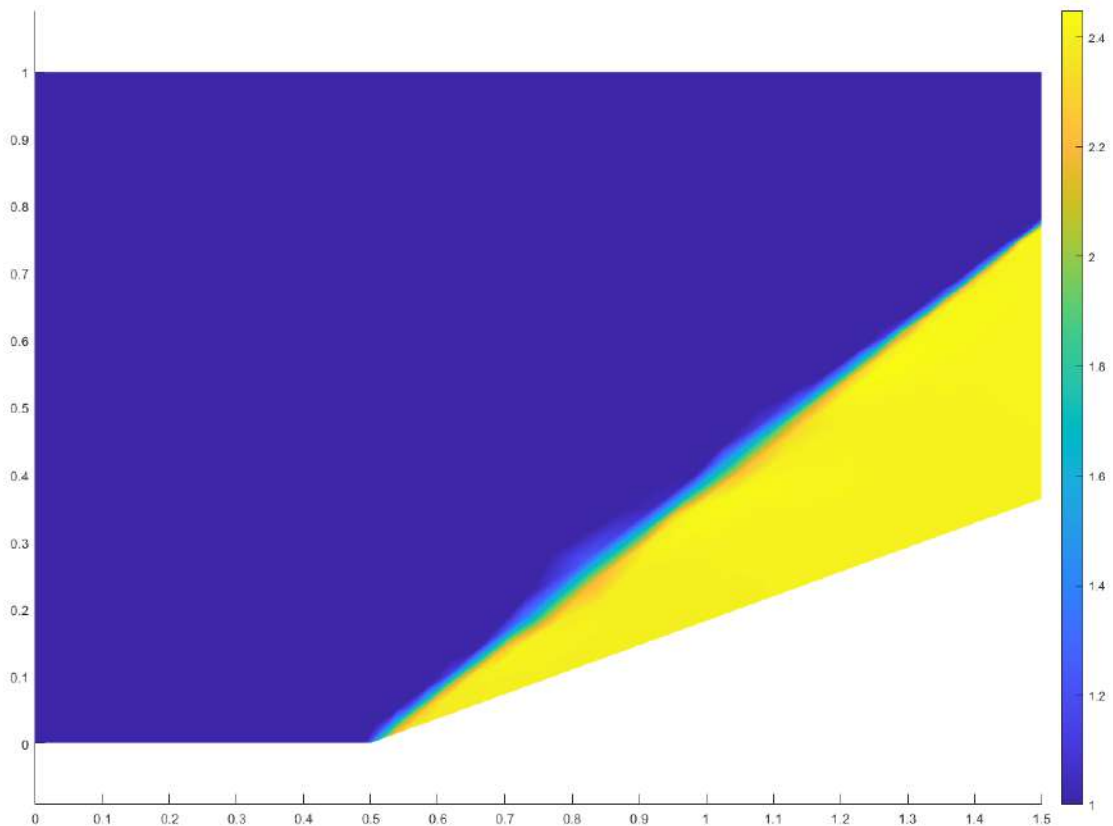

Figure 62: Flow over Wedge Problems Final Adaptive Mesh Density Plot (result)

The exact solution of the flow over wedge with an angle of θ can be computed by first calculating the shockwave angle β by solving:

$$tan(\theta) = \frac{2cot(\beta)(M_0^2 sin^2(\beta) - 1)}{2 + M_0^2(\gamma + cos(2\beta))} \qquad (1)$$

which is a transcendental equation. At which the shockwave angle β can be used to determined to density of the flow after the shockwave line from

$$\frac{\rho}{\rho_0} = (\frac{(\gamma + 1)M_0^2 sin^2\beta}{2 + (\gamma - 1)M_0^2 sin^2\beta}) \qquad (2)$$

The locations of the shockwave in the horizontal direction are then calculated on the basis of the problem geometry and the shockwave angle obtained earlier.

The comparison among meshes in each stages are shown in figure 63. Starting from initially generated mesh to 3rd mesh, it can be seen that the solution gradually approaches the value of exact solution. This implies that, by applying adaptive mesh, the solution will become closer to the exact solution.
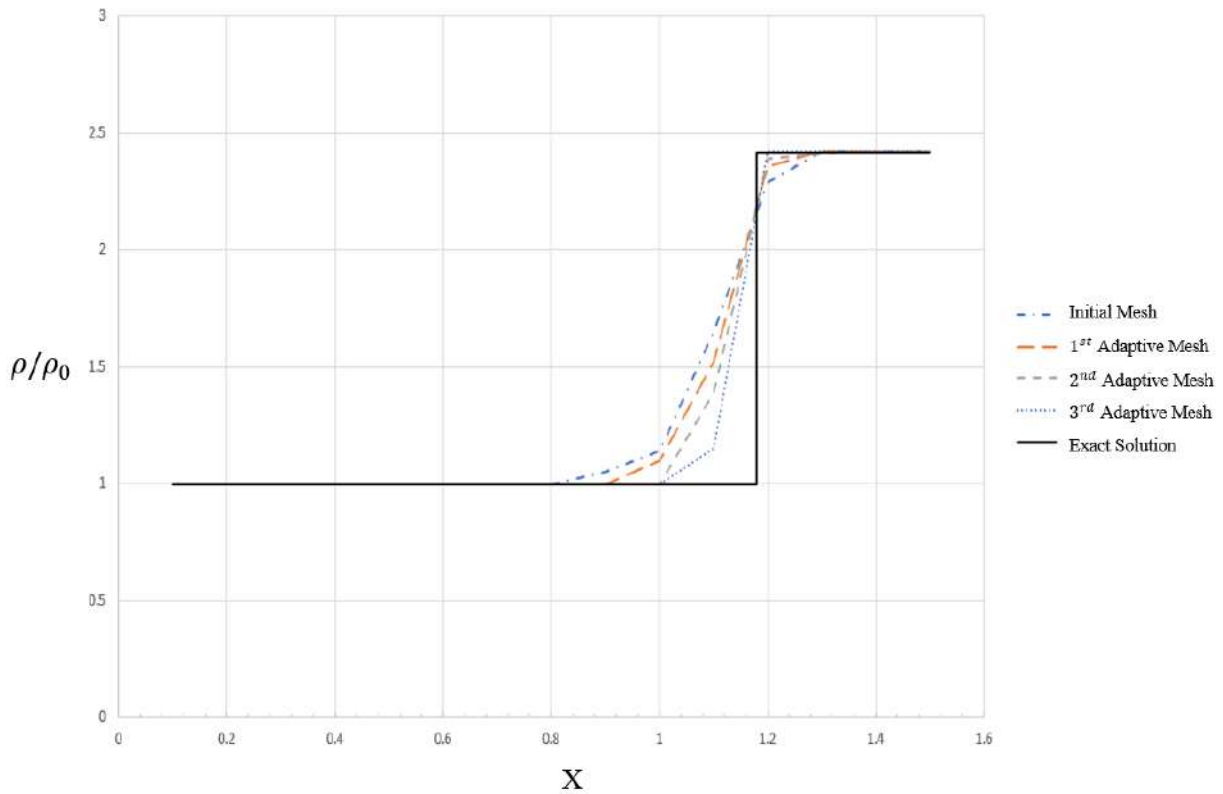

Figure 63: Flow over Wedge Problem Density Distribution Comparison Among Meshes

6.1.2 Flow Over a Diamond-shape Airfoil

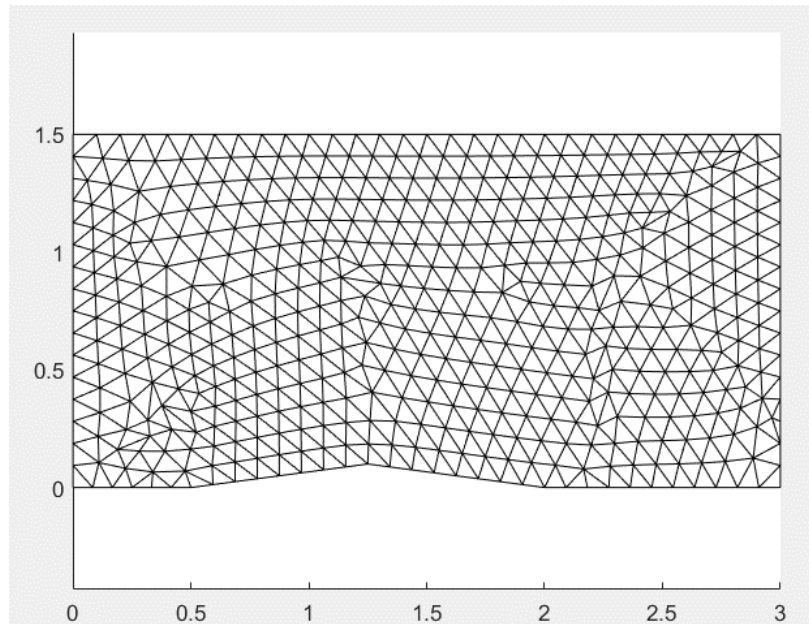The initially generated mesh of flow over diamond-shape airfoil problems can be seen from figure 64.

Figure 64: Flow over Diamond-Shape Airfoil Problems Mesh Plot (result)

The distribution of density in the flow domain solved using the originally generated mesh can be seen in figure 65. It can be observed that the incoming flows possess a uniformly distributed density. Then, when the flow reaches the starting point of the airfoil, the flow changes its direction which causes the formation of the shock wave line. From the density gradient, the density along the shock wave line drastically increased. Next, the flow travels through the maximum inclination of the airfoil where the flow's direction is changed to be along the airfoil, which causes the flow expansion. Moreover, the density along this area is decreased and causes another formation of the shock wave line starting at the maximum camber of the airfoil. After that, the flow passes the end of the airfoil where the flow is compressed. The compression of the flow causes another shock wave line. However, it can be seen that the plot does not accurately capture the sudden change in density and the change in flow direction that can be observed in the area where a large portion of the shock wave line occurs and in the area along the maximum camber of the airfoil. The reason behind this event is the size of the elements in the mentioned area used in the calculation being too large. So, to further increase the accuracy of the calculation, the size of the elements along those areas needs to be smaller.
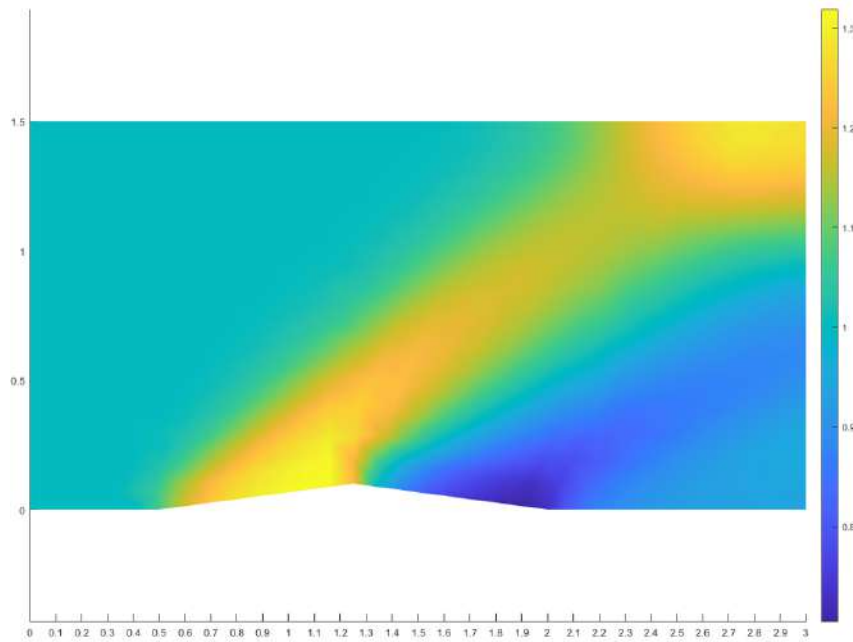
Figure 65: Flow over Diamond-Shape Airfoil Problems Density Plot (result)

After applying the adaptive meshing, the 1st Adaptive Mesh of flow over diamond-shape airfoil problems can be seen in figure 66. It can be seen that the elements within the plot have different sizes. Furthermore, the elements that reside along the shock wave line and at the leading edge of the airfoil have a smaller size compared to those of the other parts.
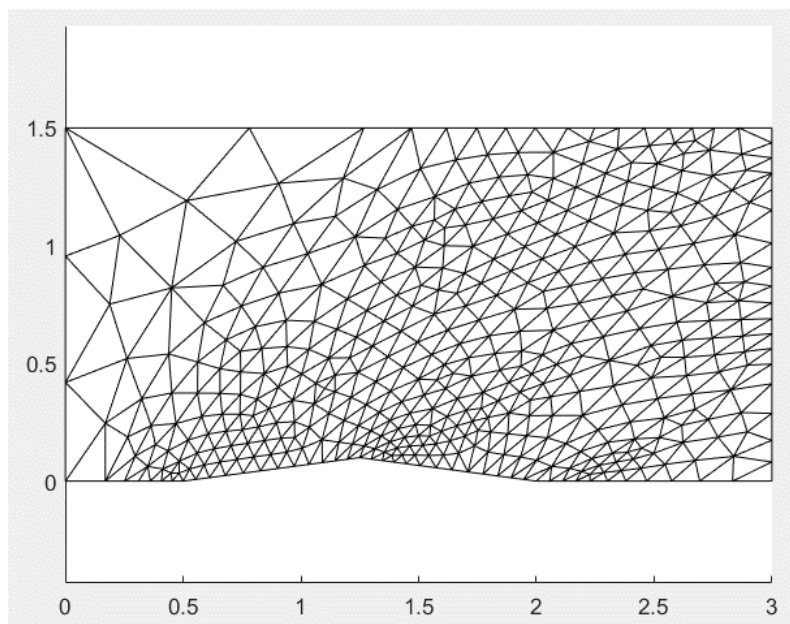

Figure 66: Flow over Diamond-Shape Airfoil Problems 1st Adaptive Mesh (result)

Using the newly generated mesh, the density plot can be determined as shown in Figure 67. Compared to the result ofto the result of the figure 65, it can be seen that the shock wave line at the maximum camber of the airfoil becomes thinner and more distinct.
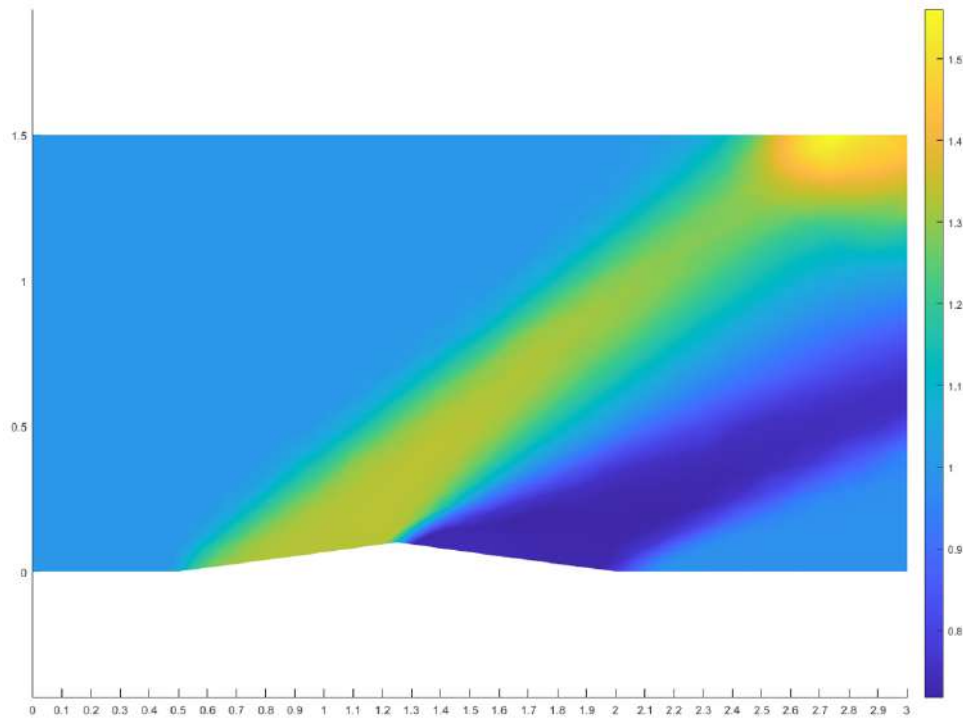
Figure 67: Flow over Diamond-Shape Airfoil Problems 1st Adaptive Mesh Density Plot (result)

After applying adaptive meshing, the 2nd Adaptive Mesh of flow over diamond-shaped airfoil problems can be seen in figure 68. It can be seen that the elements residing along the shock wave line possess even smaller size compared to the previous plot (figure 66). Moreover, it can be seen that the elements reposition themselves along the shock wave line.
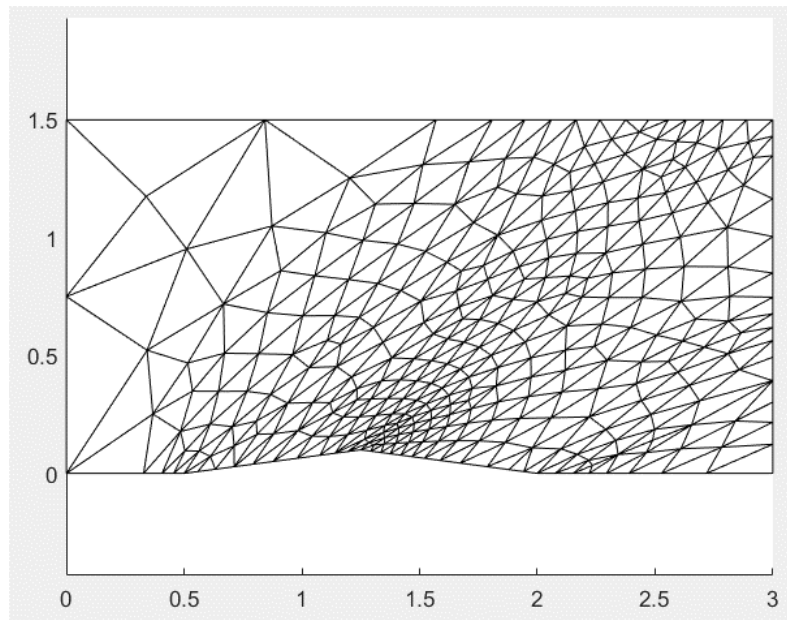

Figure 68: Flow over Diamond-Shape Airfoil Problems 2nd Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in Figure 69. Compared to the result in figure 67, it can be seen that the appearance of the shock wave line does not have drastic changes. So, further adaptive meshings are needed to be employed.
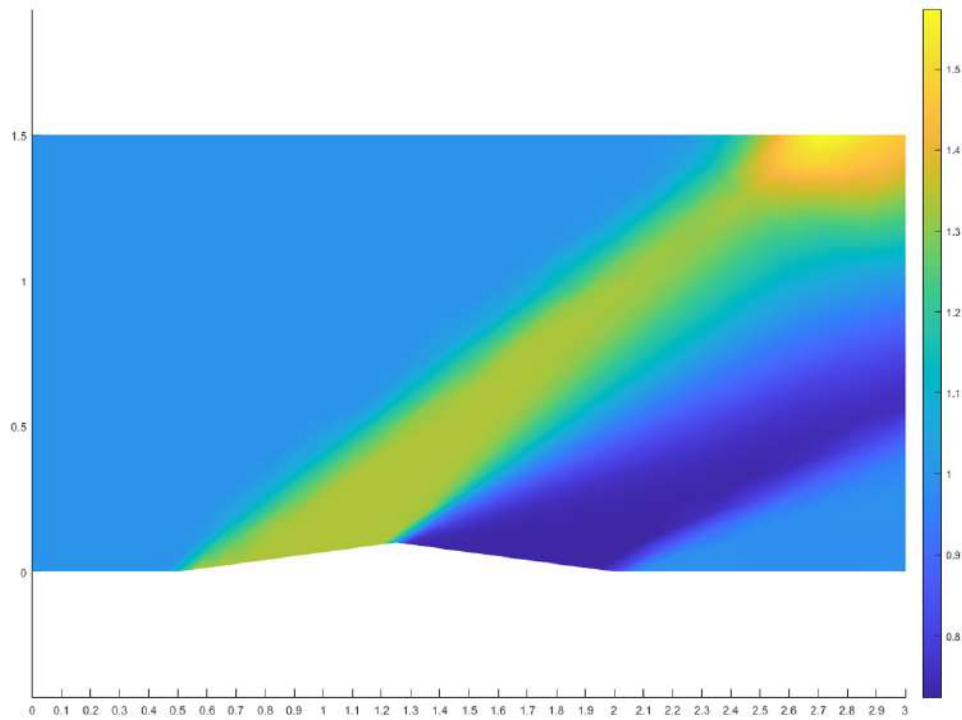
Figure 69: Flow over Diamond-Shape Airfoil Problems 2nd Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 3rd Adaptive Mesh of flow over diamond-shape airfoil problems can be seen in figure 70. It can be observed that the elements have re-positioned themselves to be more along with the shock wave line . Furthermore, small elements are present in the area along the shock wave line at the leading edge, the maximum camber and the trailing edge of the airfoil.
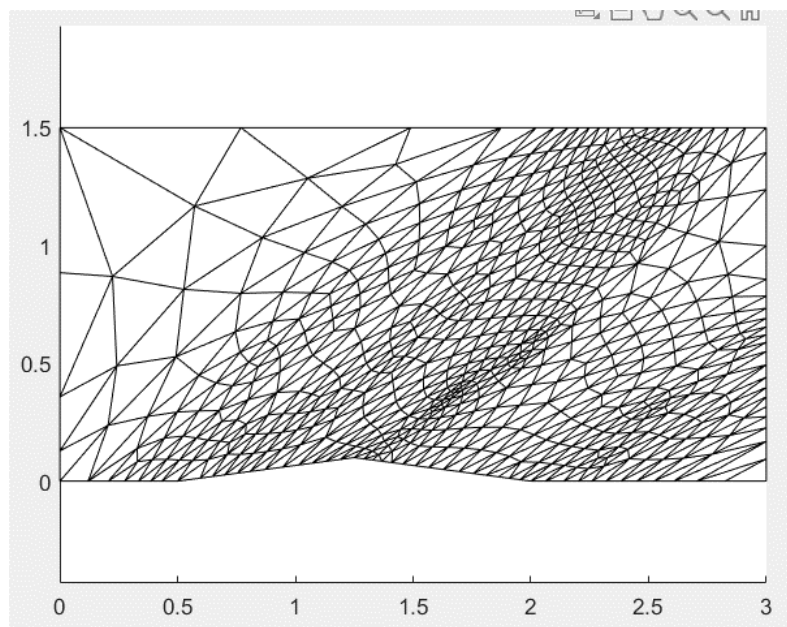

Figure 70: Flow over Diamond-Shape Airfoil Problems 3rd Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in Figure 71. Compared to the result of the figure 69, it can be seen that the shock wave lines at the leading edge, maximum camber, and the trailing edge of the airfoil are more well defined than before.
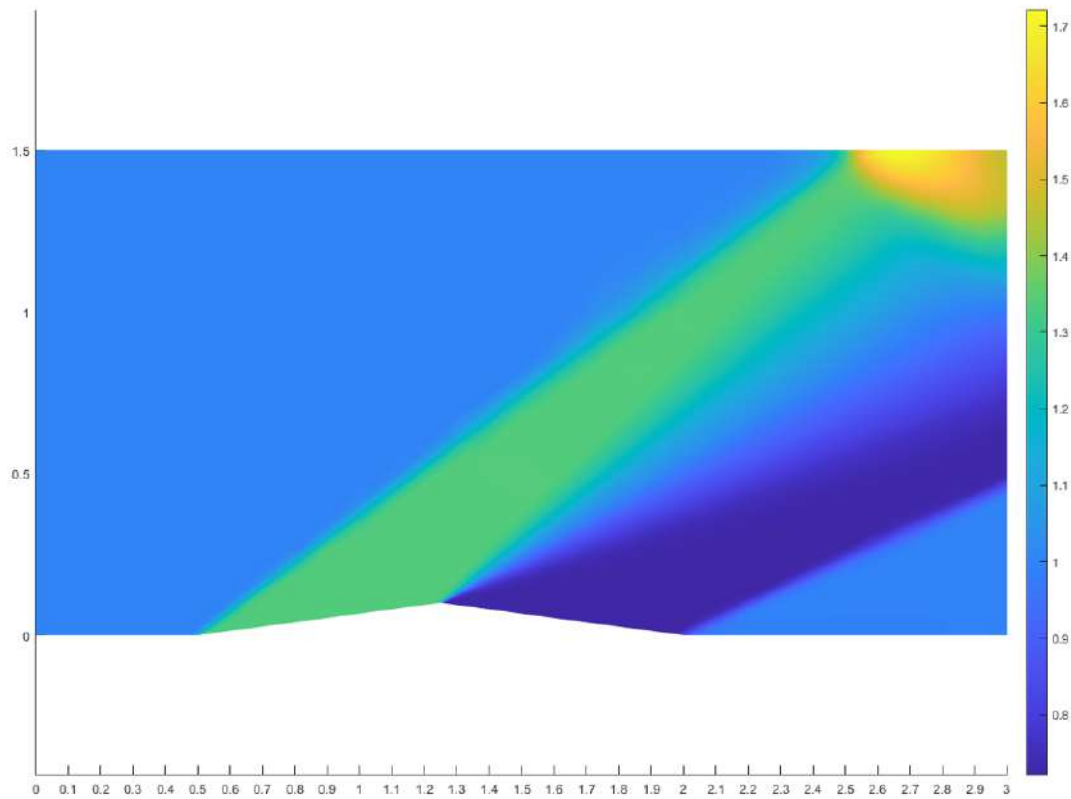
Figure 71: Flow over Diamond-Shape Airfoil Problems 3rd Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 3rd Adaptive Mesh of flow over wedge problems can be seen in figure 72. It can be observed that the elements have repositioned themselves to be more along with the shock wave line. In addition, small elements are present in the area along the shock wave line at the leading edge, maximum camber, and the trailing edge of the airfoil. Furthermore, the areas between the shock wave lines possess a larger element size compared to those along the shock wave lines.
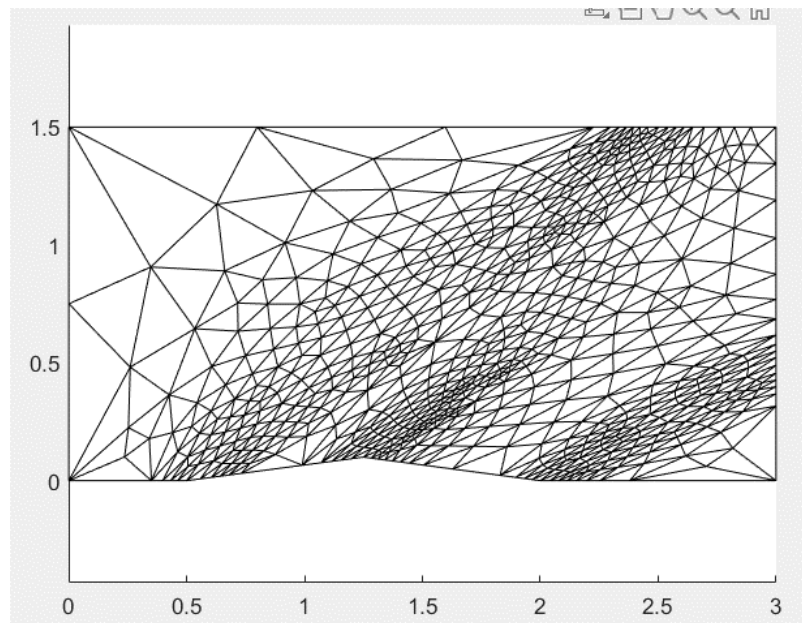

Figure 72: Flow over Diamond-Shape Airfoil Problems 4th Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in Figure 73. Compared to the result in figure 71, it can be seen that the shock wave lines at the leading edge,

65

maximum camber, and the trailing edge of the airfoil are similar to that of the previous result.
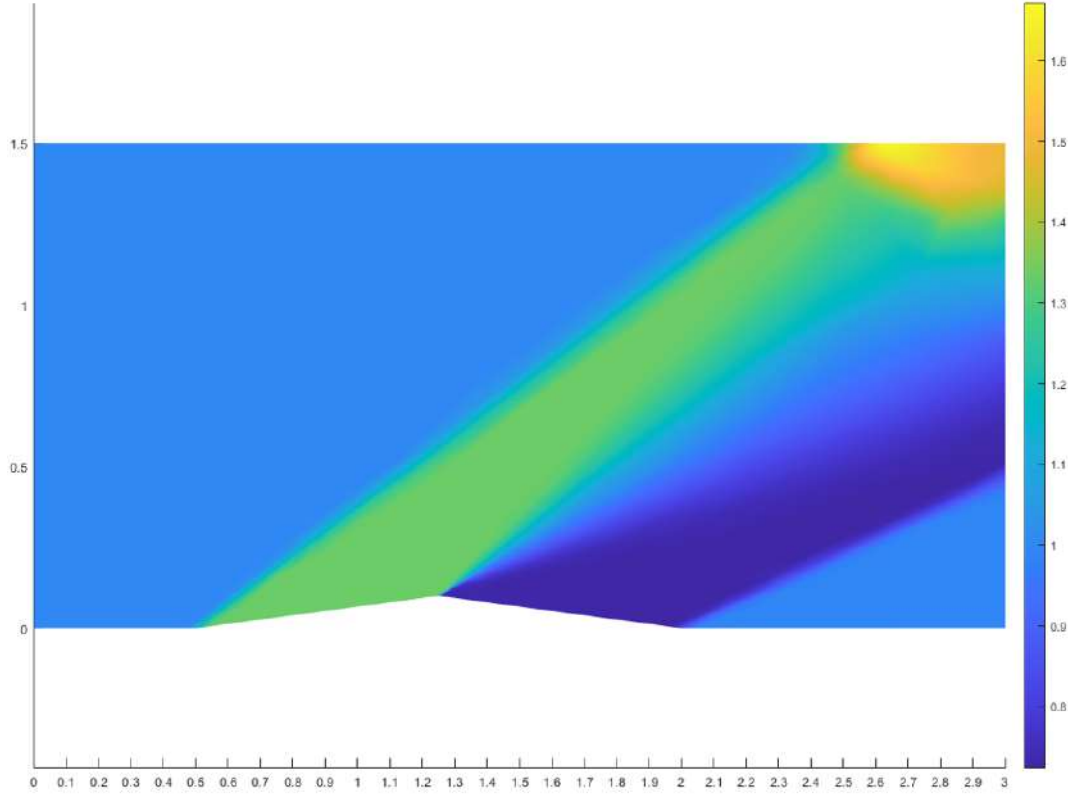

Figure 73: Flow over Diamond-Shape Airfoil Problems 4th Adaptive Mesh Density Plot (result)

The exact solution of the flow over the diamond-shaped airfoil which is actually consisting of oblique shockwave and Prandtl-Meyer expansion flow. The exact solution for an oblique shockwave can be found with a wedge angle of δ can be computed by first calculating the shockwave angle β by solving:

$$tan(\delta) = \frac{2cot(\beta)(M_0^2 sin^2(\beta) - 1)}{2 + M_0^2(\gamma + cos(2\beta)}$$ (3)

At which the shockwave angle β can be used to determined to density of the flow after the shockwave line from

$$\frac{\rho}{\rho_0} = (\frac{(\gamma + 1)M_0^2 sin^2\beta}{2 + (\gamma - 1)M_0^2 sin^2\beta})$$ (4)

Then, we need to find the flow Mach number directly after the shockwave in order to calculate the exact solution of the expansion flow from the maximum camber region of the airfoil. This Mach number can be determined from:

$$M_1^2 sin^2(\beta - \delta) = \frac{M_0^2 sin^2\beta + 2/(\gamma + 1)}{2\gamma M_0^2 sin^2\beta/(\gamma - 1) - 1}$$ (5)

Once we obtain M_1 from the equation above, we can find the exact solution of the expansion flow by calculating the forward wave angle θ_1 by

$$\theta_1 = \sqrt{\frac{\gamma+1}{\gamma-1}} tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}(M_1^2 - 1)} - tan^{-1}\sqrt{M_1^2 - 1} \tag{6}$$

Then determine the rearward wave angle from

$$\theta_2 = \theta_1 + \xi \tag{7}$$

where ξ is the angle of the deflection in the maximum camber region. We can then find the Mach number of the flow after rearward wave angle by solving the equation:

$$\theta_2 = \sqrt{\frac{\gamma+1}{\gamma-1}} tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}(M_2^2 - 1)} - tan^{-1}\sqrt{M_2^2 - 1} \tag{8}$$

At this point, we can then find the density ratio ρ_2/rho_1 using the Mach number before the forward wave angle M_1 and after the rear wave angle M_2. The relation is as follow:

$$\frac{\rho_2}{\rho_1} = (\frac{1 + ((\gamma-1)/2)M_1^2}{1 + ((\gamma-1)/2)M_2^2})^{\frac{1}{\gamma-1}} \tag{9}$$

The calculation for an aft shockwave follow the same relation as discussed earlier where the initial Mach number is now the flow Mach number after rear wave angle M_2 instead. The location of the shockwave and wave angle are calculated based on the obtained angle and problem geometry.

The comparison among meshes in each stages are shown in figure 74. Starting from initially generated mesh to 4th mesh, it can be seen that the solution gradually approaches the value of exact solution. This implies that, by applying an adaptive mesh, the solution will become closer to the exact solution.
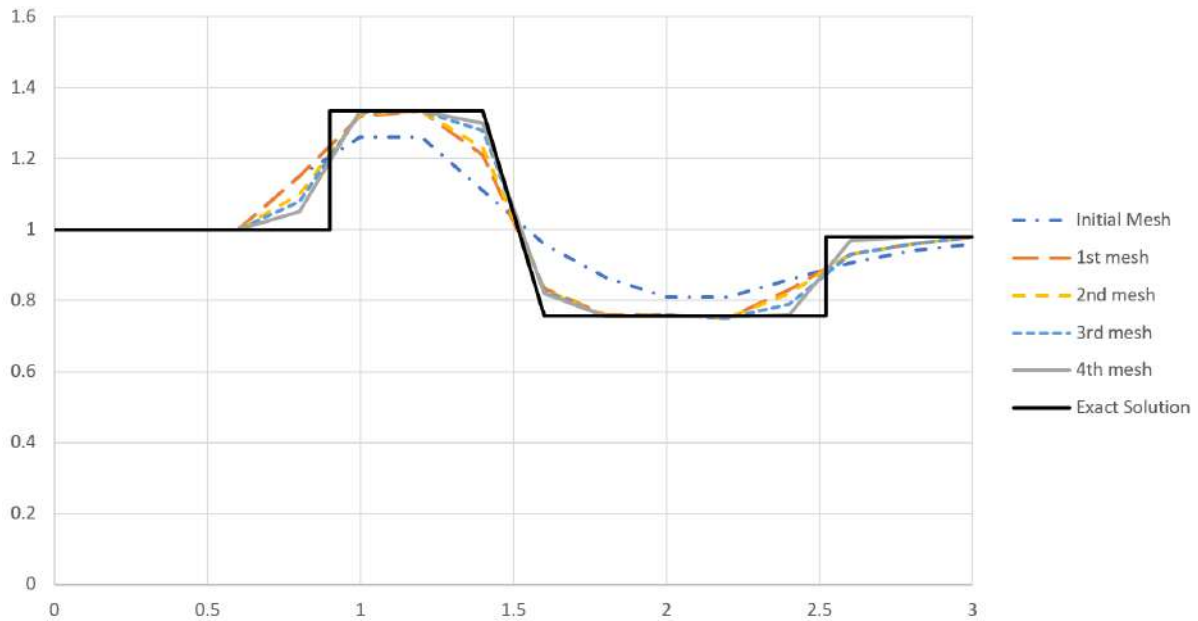

Figure 74: Flow over Diamond-Shape Airfoil Problem Density Distribution Comparison Among Meshes

6.1.3 Expansion Flow

The initially generated mesh of the expansion flow problem can be seen in Figure 75.
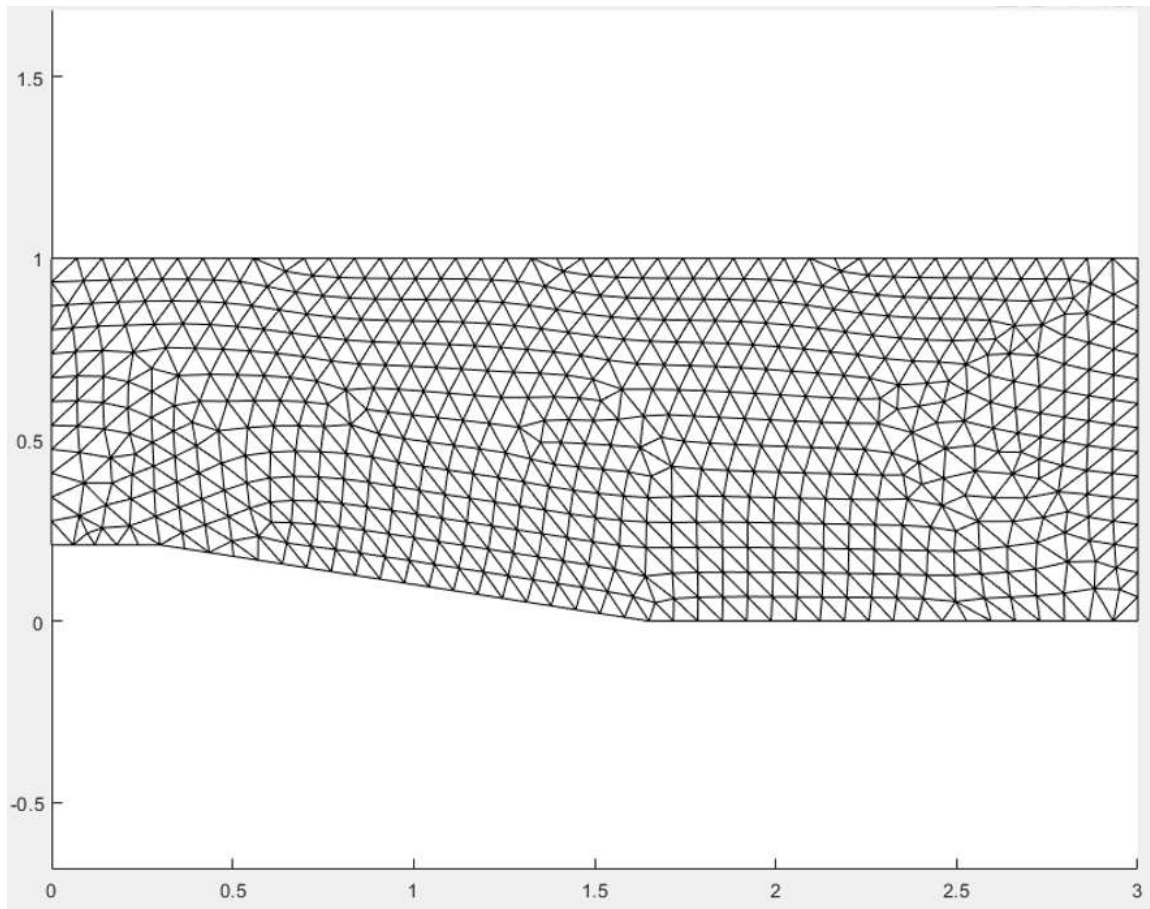

Figure 75: Expansion Flow Problem Mesh Plot (result)

The distribution of density in the flow domain solved using the originally generated mesh can be seen in figure 76. From the density gradient, it can be observed that the incoming flow possesses a uniformly distributed density. As the flow moves to the corner, the flow started to expand, which gradually reduces the density. Moreover, it can be seen that the shock wave has low accuracy which can be observed at the corner. This is because the mesh used in the gradient generation possesses the element that is too large around the corner. So, to increase the accuracy, further adaptive meshings are needed to decrease the element's size around the corner.
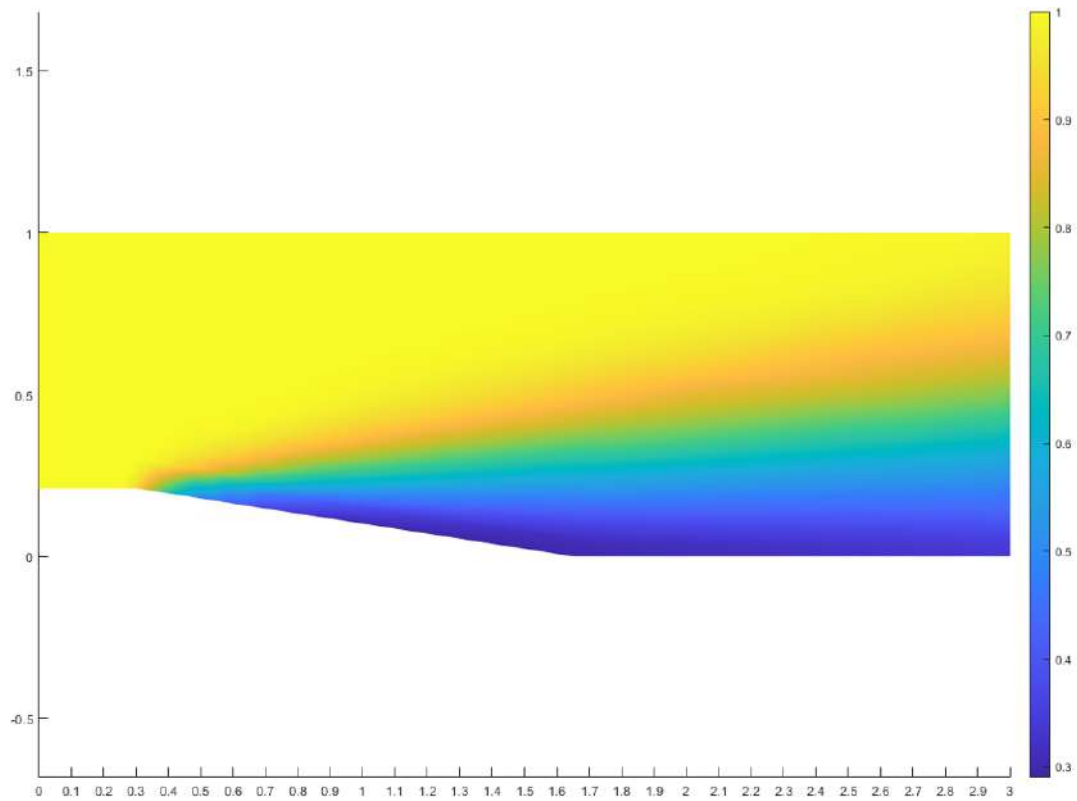
Figure 76: Expansion Flow Problem Density Plot (result)

After applying the adaptive meshing, the 1st Adaptive Mesh of expansion flow problems can be seen in figure 77. It can be seen that the elements within the plot have different sizes. Furthermore, the elements that reside around the shock wave line and the corner have a smaller size compared to those of the other parts.
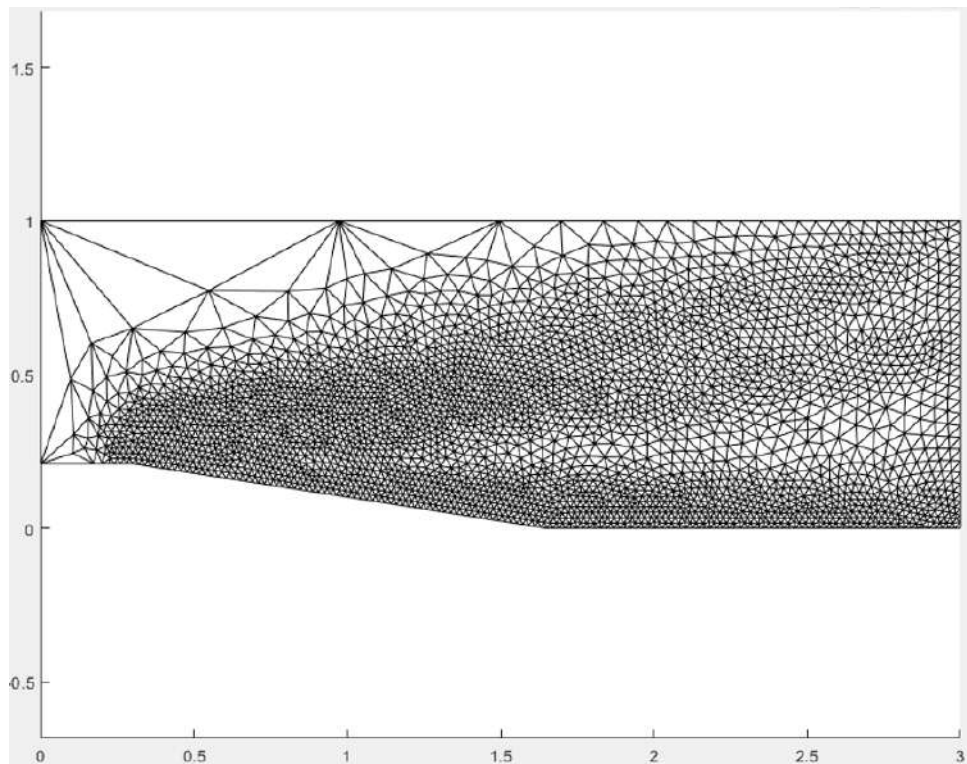

Figure 77: Expansion Flow Problem 1st Adaptive Mesh (result)

Using the newly generated mesh, the density plot can be determined as shown in Figure 78. Compared to the result of the figure 76, it can be seen that the shock wave line generated from the corner becomes more accurate than the previous result.
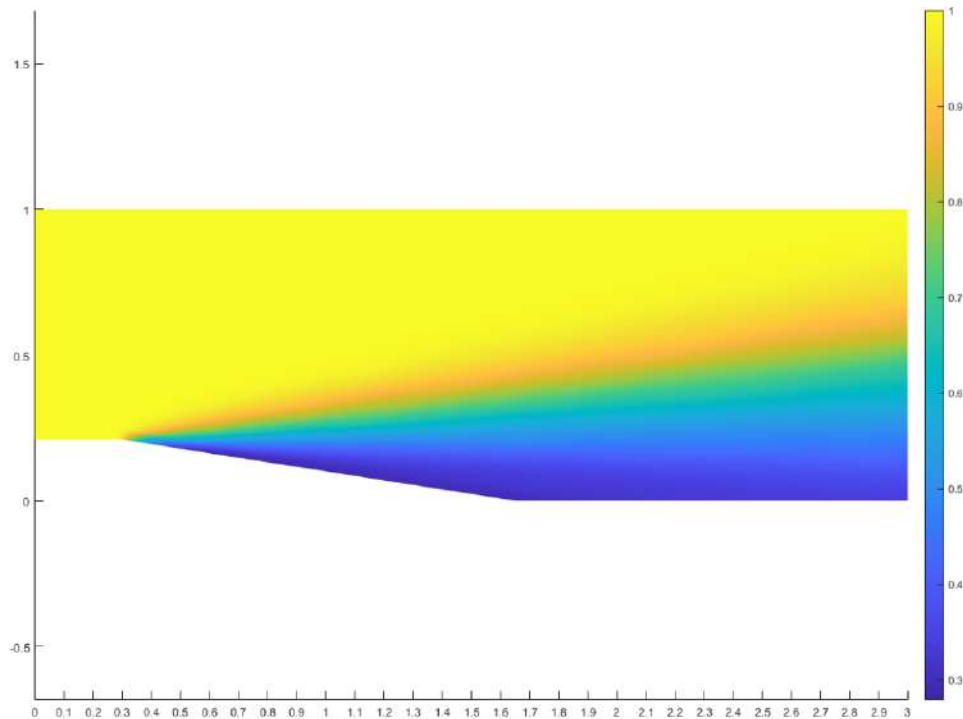

Figure 78: Expansion Flow Problem 1st Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 2nd Adaptive Mesh of expansion flow problems can be seen in figure 79. It can be seen that the elements residing along the shock wave line and the corner possess even smaller size compared to the previous plot (figure 77). Moreover, it can be seen that the elements around the back of the flow domain become larger than that of the previous result.
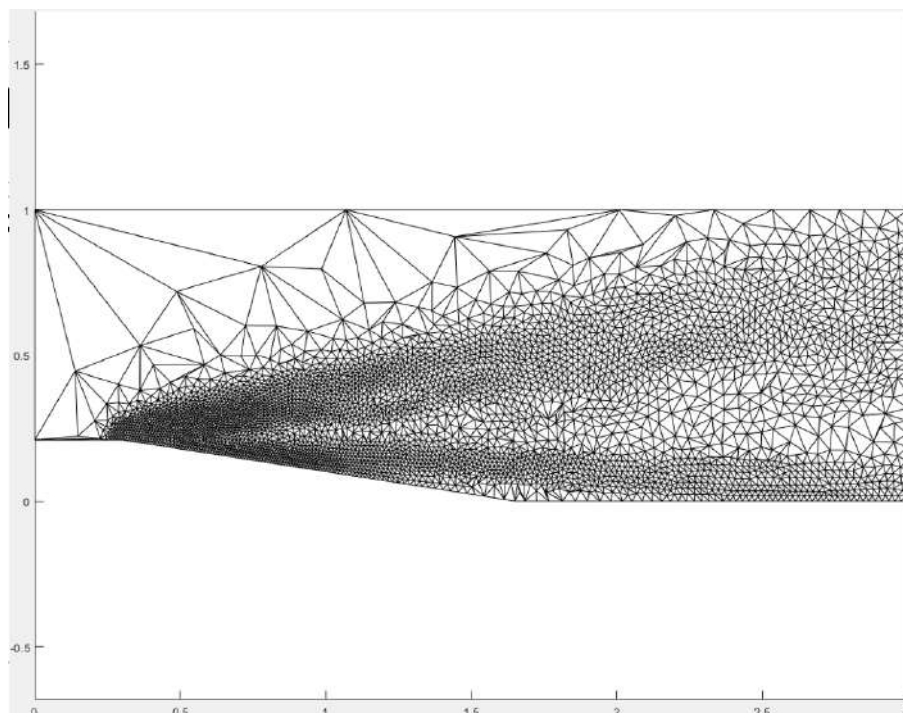

Figure 79: Expansion Flow Problem 2nd Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in figure 80. Compared with the result of figure 76, it can be seen that the shock wave line generated from the corner becomes even slimmer.

Figure 80: Expansion Flow Problem 2nd Adaptive Mesh Density Plot (result)

After applying the adaptive meshing, the 3rd Adaptive Mesh of expansion flow problem can be seen in figure 81. It can be seen that the large number of small-size elements are presented around the corner where large elements are presented in other areas.

Figure 81: Expansion Flow Problem Final Adaptive Mesh Plot (result)

Using the newly generated mesh, the density plot can be determined as shown in figure 82. Compared with the result of figure 80, it can be seen that the sh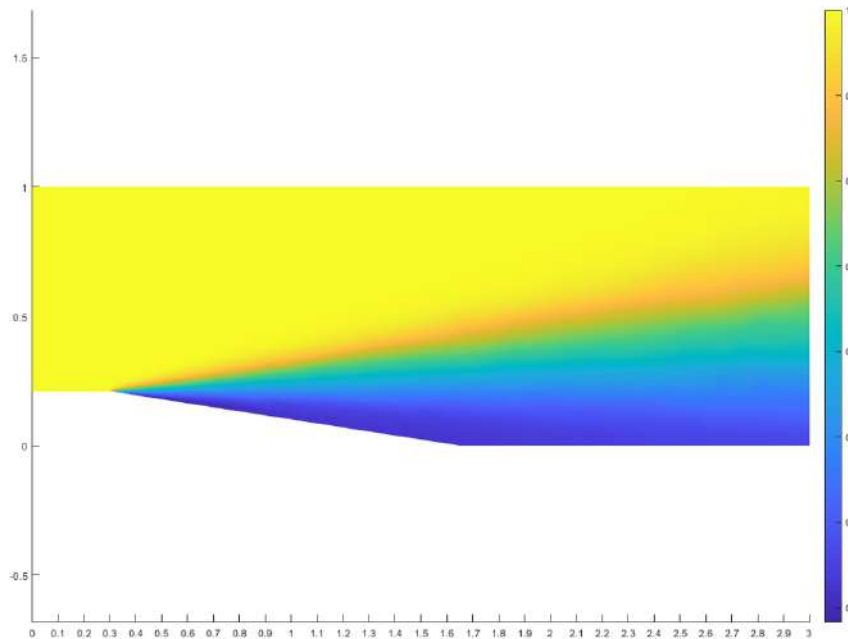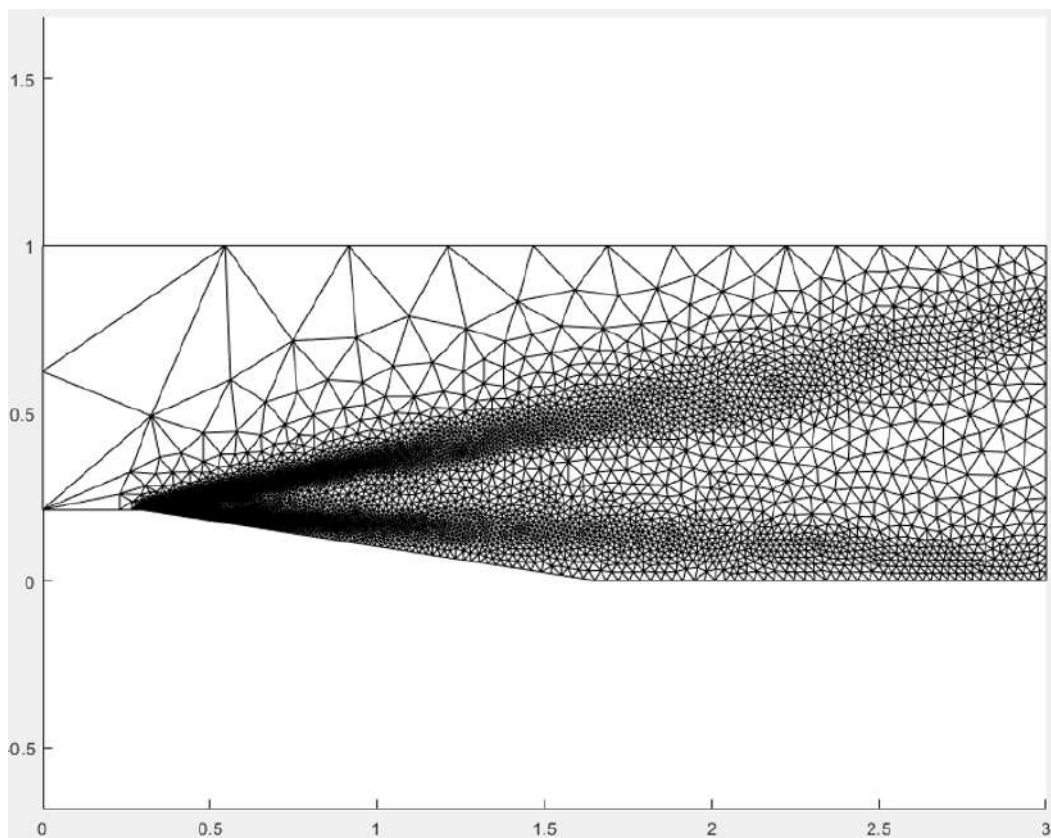ock wave line generated from the corner becomes even slimmer than before. Moreover, the change in density can be clearly seen from the gradient compared to the previous results.
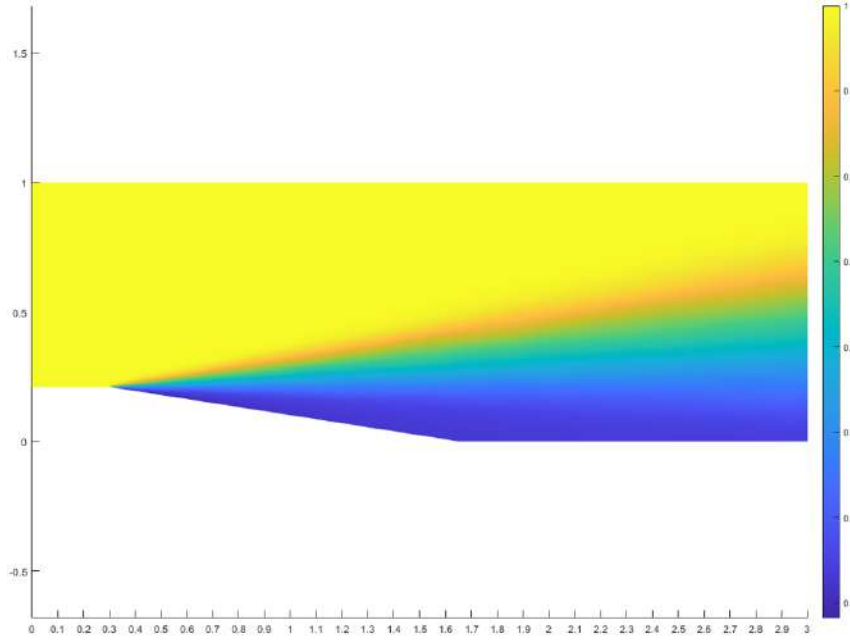

Figure 82: Expansion Flow Problem Final Adaptive Mesh Density Plot (result)

Again, we can find the exact solution of the expansion flow by calculating the forward wave angle θ_1 by

$$\theta_1 = \sqrt{\frac{\gamma+1}{\gamma-1}} tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}(M_1^2-1)} - tan^{-1}\sqrt{M_1^2-1} \tag{10}$$

Then determine the rearward wave angle from

$$\theta_2 = \theta_1 + \xi \tag{11}$$

where ξ is the angle of deflection of the geometry. We can then find the Mach number of the flow after rearward wave angle by solving the equation:

$$\theta_2 = \sqrt{\frac{\gamma+1}{\gamma-1}} tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}(M_2^2-1)} - tan^{-1}\sqrt{M_2^2-1} \tag{12}$$

At this point, we can then find the density ratio ρ_2/rho_1 using the Mach number before the forward wave angle M_1 and after the rear wave angle M_2. The relation is as follow:

$$\frac{\rho_2}{\rho_1} = (\frac{1+((\gamma-1)/2)M_1^2}{1+((\gamma-1)/2)M_2^2})^{\frac{1}{\gamma-1}} \tag{13}$$

For an expansion flow problem, the hypothetical lines are instead measured from bottom to top instead. As the wave angle are relatively shallow and as such a horizontal measurement line would only pass one expansion wave. So, a vertical measurement at x = 2 is used instead to compare the accuracy. The comparison among meshes in each stages are shown in figure 83. From the initially generated mesh to 3rd mesh, it can be seen that the solution gradually approaches the exact solution value. This implies that, by applying an adaptive mesh, the solution will become closer to the exact solution.
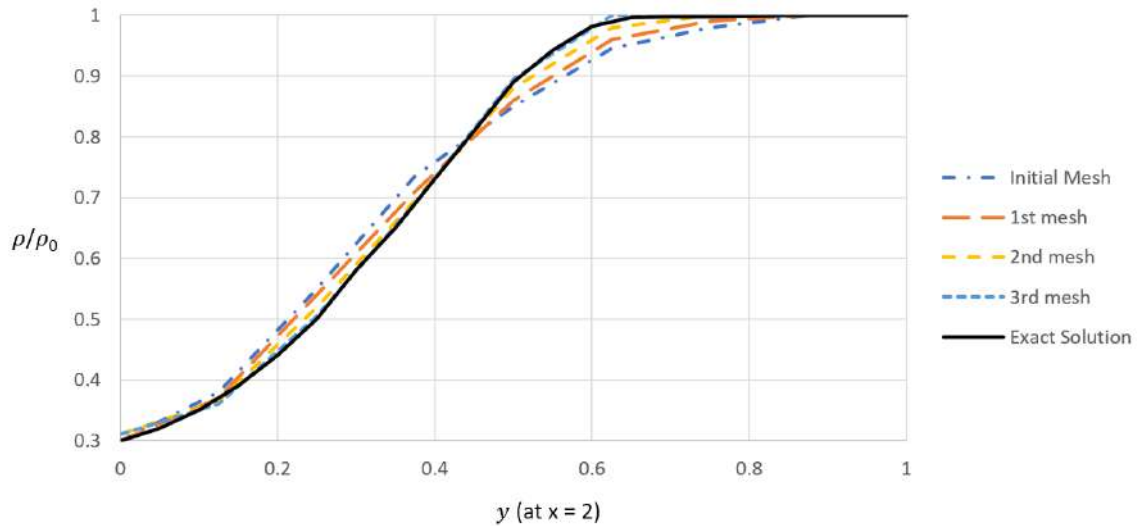
Figure 83: Expansion Flow Problem Density Distribution Comparison Among Meshes

6.2   Discussion

From figure 63, 74, and 83, it can be observed that the flow solution from the three problems (flow over wedge, flow over a diamond-shape airfoil, and expansion flow) show similar trends. The graphs convey the comparison among meshes in each different stages starting from the initially generated mesh to its ultimately generated mesh. It can be seen that the solutions gradually approach the value of the exact solution each time the adaptive mesh has been applied.

Moreover, since the developed software was built with the requirement that the user will interact with the software through CLI [9], the user experience of the user with relatively weak programming background could be different from those with a strong programming background. This issue can cause the user to be turned off by the complexity in using the software, which means the user will not be able to utilize the developed software at its utmost capabilities. However, this issue can be tackled by two solutions: a well-written document and guide on how to operate the software and an improvement of user interface which will be further discussion in section 7.2.

Furthermore, other limitation of the software is in the gradient plots. The plot is generated from MATLAB API trisurf (T, x, y, z) [13] which plots the 3-D triangular surface defined by the points in vectors x , y , and z , and a triangle connectivity matrix T . Then, the 3-D is set in a certain view such that the 2-D gradient plot can be observed. However, with this approach, the data within the plot cannot be accessed and displayed using a normal MATLAB data tips tool [14] that is used to select and display the value of the graph at a coordinate of interest. The work around of this issue is to use a color picker to pick the color data from the area of interest and match it with the color map generated from MATLAB's API [1]. Obviously, this workaround works, but it is not efficient for the user since the user has to manually extract the data, which can be very tedious. Therefore, the solution to this problem is to find another way to plot the gradient without using trisurf(T,x,y,z) function.

7 Conclusion

7.1 Assessment

In this project, the adaptive remeshing software is developed to automatically improve the mesh quality based on previous flow behaviour results from the program hiflow . The goal of this project is to redevelop software based on the old Fortran program, REMESH , in a newer programming language, MATLAB, and add compatibility with the program hiflow such that the user can easily display a new and improved mesh. The newly generated mesh will greatly increase the accuracy of the problem result. The redeveloped software will also allow the future student to explore and visualize various complex flow interactions, which will greatly help in their learning of the Aerodynamic II course. In this project, this program is used to solve an array of problems: flow over a wedge, flow over a cylinder, flow over a diamond-shaped airfoil, and expansion flow. The results show that the calculated solution of the density distribution at certain sections in the flow domain become more accurate (approaching the exact solution) when the final adaptive mesh is used in the calculation. Moreover, with MATLAB's API [1], the solution can be displayed in the form of plots and gradients, which greatly makes the user experience more interactive and flawless. However, since the user can only interact with the software through CLI (Command Line Interface), some users with weak programming backgrounds might find it understandably hard or unintuitive to navigate through the software. Moreover, the gradient plotted by the software does not have a data tips tool (like that of the usual MATLAB's plot [15]) where it shows the user values at the point of interest. Therefore, it can be challenging to extract data from the gradient plot when the user needed it.

7.2 Next Steps

In section 7.1, there are two main issues in the software: user experience with using the CLI and data acquisition from the gradient plot. The first issues can be tackled using various solutions. It can be solved by optimizing the existing CLI to make it as fluid as possible to navigate through the software. However, the approach will surely reach its limit since the CLI inherently does not have many design possibilities that can really make a drastic difference. Therefore, the actual solution is to build a GUI (Graphic User Interface) [16] for the user to interact with the software. However, with the amount of time and manpower provided, it was not possible to do such a task since it was too complicated and outside of the project's scope. For the data acquisition problem from the gradient plot, since the gradient was generated using MATLAB's API, there is not much that the team can do. So, the alternative is to use a color picker to pick the color data from the area of interest and match it with the color map generated from the MATLAB API. The solution to this problem is to find another way to plot the gradient without using trisurf(T,x,y,z) function [13] which can be further looked up in MATLAB file exchange web site [17]. Lastly, to further prove the capabilities of the software, the software must be tested with more complex supersonic flow problems under various conditions.

References

[1] The MathWorks Inc. Graphics. Available at `https://www.mathworks.com/help/matlab/graphics.html`.

[2] SimScale. Computational fluid dynamics: Cfd software. Available at `https://www.simscale.com/product/cfd/`.

[3] P. Dechaumphai. Compressible Fluid Flow. 2021.

[4] Elsevier B.V. Finite volume method. Available at `https://www.sciencedirect.com/topics/engineering/finite-volume-method`.

[5] Randall L. Finite volume methods for hyperbolic problems, cambridge university press, 2002.

[6] Mastin C. W. Thompson J F., Warsi Z. U. A. Numerical grid generation: Foundations and applications, north-holland, elsevier.

[7] SimScale. What is cfd (computational fluid dynamics)? https://www.simscale.com/docs/simwiki/cfd-computational-fluid-dynamics/what-is-cfd-computational-fluid-dynamics/.

[8] A.T. Patera J. Peraire. Advances in Adaptive Computational Methods in Mechanics. In *Studies in Applied Mechanics*, 1998.

[9] W3Schools. Command line interface. Available at `https://www.w3schools.com/whatis/whatis_cli.asp`.

[10] The MathWorks Inc. Command window. Available at `https://www.mathworks.com/help/matlab/ref/commandwindow.html`.

[11] Mehmet Oezcan. Convergence of a cfd simulation. Available at `https://www.simscale.com/knowledge-base/how-to-check-convergence-of-a-cfd-simulation/`.

[12] 2002-2022 Dassault Systèmes SolidWorks Corporation. Solidworks. Available at `https://www.solidworks.com/`.

[13] The MathWorks Inc. trisurf. Available at `https://www.mathworks.com/help/matlab/ref/trisurf.html`.

[14] The MathWorks Inc. datatip. Available at `https://www.mathworks.com/help/matlab/ref/matlab.graphics.datatip.datatip.html`.

[15] The MathWorks Inc. Matlab plots. Available at `https://www.mathworks.com/help/matlab/creating_plots/types-of-matlab-plots.html`.

[16] The MathWorks Inc. Matlab gui. Available at `https://www.mathworks.com/discovery/matlab-gui.html`.

[17] The MathWorks Inc. Matlab file exchange. Available at `https://www.mathworks.com/matlabcentral/fileexchange/`.

Appendix

Redeveloped Adaptive Meshing Software Code

The code is be accessed via this url:

```
https://github.com/T-O-A-D/Remesh-Release
```

Fully Labeled Input Files

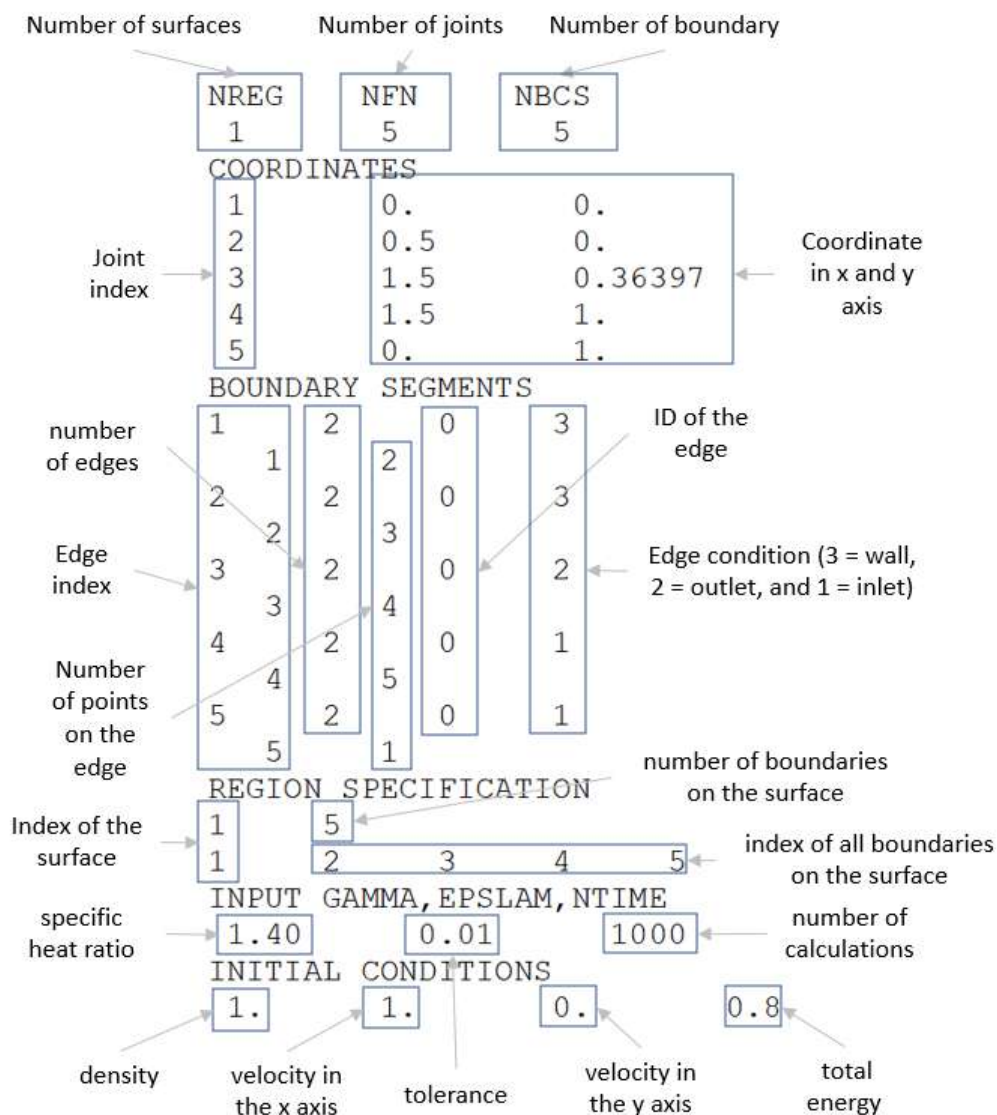Fully labeled input file (WEDG.FIX) (figure 84)

Figure 84: Fully labeled input file (WEDG.FIX)
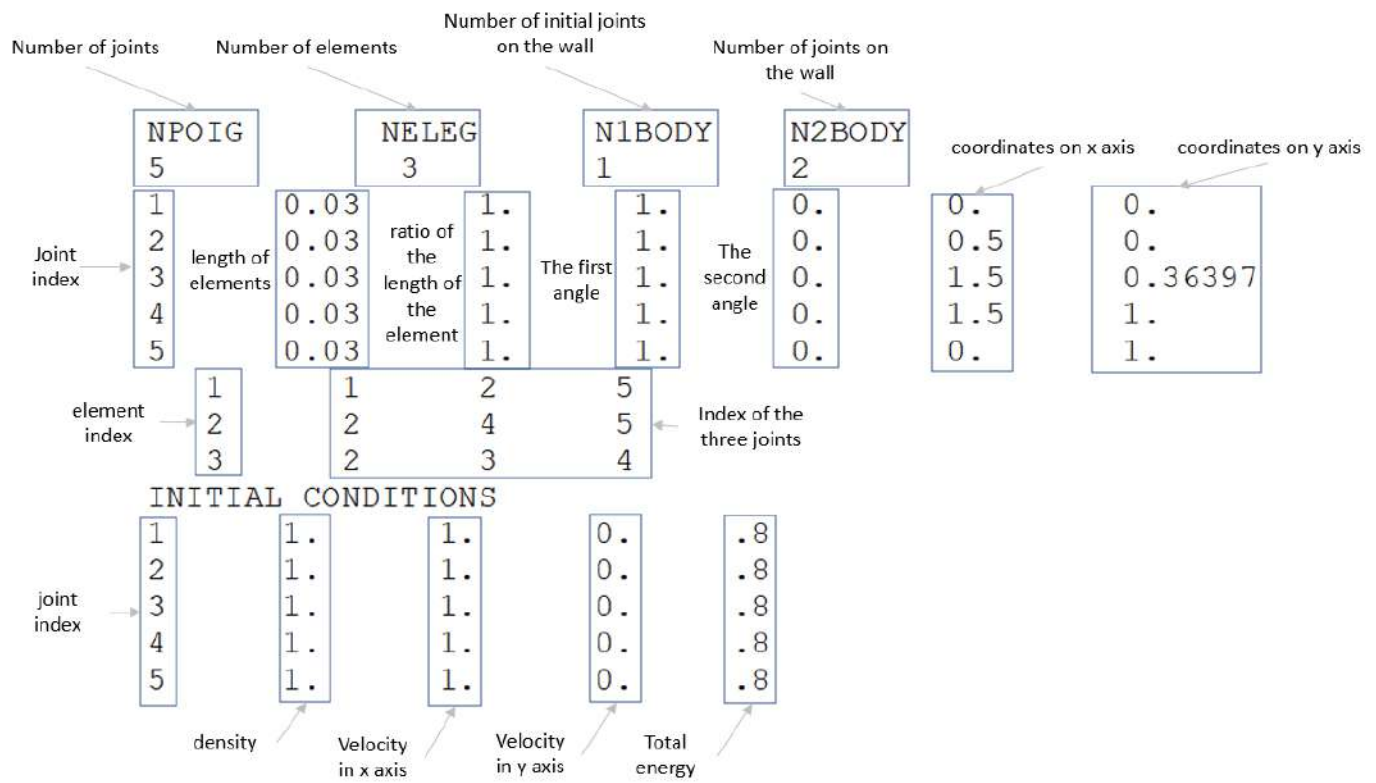
Fully labeled input file (WEDG.RE1) (figure 85)

Figure 85: Fully labeled input file (WEDG.RE1)