Energy research Centre of the Netherlands
and
Delft University of Technology

Validation of AVATAR Airfoils using SU2 code

by

Pratik Nayak
supervised by
Dr. Hüseyin Özdemir, Wind Energy Group,
Energy research Center of the Netherlands
and
Dr. Duncan Van der Heul, Numerical Analysis Group,
Applied Mathematics Department, TU Delft

A project report submitted in partial fulfillment for the
Honors Programme Master
in
Fluid Mechanics
at the
Mechanical, Maritime and Materials Engineering Department, TU Delft

November 25, 2016

"God put me on this earth to accomplish a certain number of things. Right now, I am so far behind, I will never die."

Calvin from Calvin and Hobbes by Bill Watterson.

"The human brain is finite. The powers of its imagination, infinite."

Acknowledgements

I would like to thank Dr. Hüseyin Özdemir for giving me an opportunity to work on this project and for all the advice and guidance he provided throughout the project. I would also like to thank my supervisor at TU Delft, Dr. Duncan van der Heul for his valuable insights throughout the project. I would like to thank Giridhar Ramanujam and Akshay Koodly Ravishankara for helping me understand some aspects of Aerodynamics, a field, in which, I clearly have a lot to learn.

Last but not the least, I would like to thank the Delft Wind Energy Institute (DUWIND) for giving me an opportunity to study at TU Delft by supporting me through their scholarship. I am extremely greatful to them.

Contents

List of Figures

List of Tables

Abbreviations

SU^2 **S**tanford **U**niversity **U**nstructured

Physical Constants

PI π = 3.141592653

Symbols

Symbol	***Name***	***Unit***
α	Angle of attack	degrees
Re	Reynolds Number	
ω	angular frequency	$\mathrm{rad\,s^{-1}}$
C_l	Coefficient of lift	
C_d	Coefficient of drag	

To my parents . . .

Chapter 1

Introduction

The AVATAR [1] project is a project that aims to develop and validate aerodynamic models that can be used for the design of large scale wind turbines upto 20MW. This project was conducted as a part of the AVATAR project to validate the results from different AVATAR partners for various airfoils designed specifically for wind turbines. The aim of this project was to establish the Stanford SU2 code as a CFD tool to evaluate the airfoil properties.

1.1 Motivation

Wind Energy has been one of the most important areas of research in renewable energy in the past twenty years. Especially in Europe where the wind potential is very high. Many of the northern European countries have made use of this renewable resource to satisfy their energy needs and even have excess energy which they have sold to other countries.

As the countries and the people have become more developed, energy has become one of the basic needs and hence a lot of time and effort has been devoted to research and development of wind energy and energy in general.

Many of these efforts in wind energy have been to design efficient rotors and fluid mechanics has been one of the most important aspect which has been used to optimize the rotors. The optimization of airfoils can increase the efficiency of the wind turbines. Efforts have been directed towards two major approaches, experimental and computational. As explained later, at very high Reynolds numbers it becomes very difficult to obtain experimental results for the turbines. Hence we have to resort to computational

[1] http://www.eera-avatar.eu/

methods for our analyses. A truly computational method, based on first principles, which solves the Navier Stokes equations for the flow problem will take unrealistic amount of computational power and time which is not feasible when the flow is complicated and turbulent because the time and length scales that it has to resolve are impractical. To obtain To obtain the results in a reasonable amount of time, we resort to turbulence models which use some empirical relations and some experimental constants to close the equations.

Various methods have been developed for solving turbulent flows and as they are in some sense empirical or depend on empirical data, they can give misleading results for some flow problems. Therefore for reliability, it is necessary to test all the possible models available and choose the model suitable for the particular flow problem.

Therefore, in this report we have tried to show that the Stanford SU2 code, which is an open source CFD software and a tool which can be used to simulate the flow problems for the wind turbine applications. SU2 has the capability of airfoil design optimization and this may prove to be useful when designing and optimizing the arifoils required for the AVATAR project. These capabilities can only be used when SU2 is proved to be sufficient for cases which have been tested by the other AVATAR partners.

1.2 Background

Large scale wind turbines will be mounted at larger heights than the present turbines where they will be able to harness the higher wind speeds required to produce higher power. Hence to have a nominal tip speed ratio, we need to increase the rotational speed, ω or the radius of the turbine blades. Tip speed ratio is defined as the tangential speed of the tip of the blade which can be controlled by the rotational speed (ω) and the radius of the blade. It is one of the important performance parameters of a wind turbine. Increasing the radius of the turbine is easier than the former and hence large scale wind turbines have a large turbine blade length.

When the radius of the turbine is large and the free-stream velocity is high, this makes the Reynolds number very high and may range from 10 million to even 20 million and higher. Experimentation at this high Reynolds numbers is very difficult and very few experimental results are available. Hence we must resort to computational efforts in order to obtain reliable results for the turbines.

The SU2 code is a unstructured CFD solver which solves the partial differential equations for the required flow problem using turbulence models when required. An introduction

to the code and tutorials may be found [1] and in [2].

1.3 Goals and Objectives

The objectives of the project were the following:

1. Verify the SU2 software as a reliable CFD solver for the AVATAR project.
2. Validate the results of the AVATAR partners using the SU2 software and establish the different results obtained through different methods.
3. Establish the influence of the various mesh types and parameters on the solution accuracy and time.
4. Observe and verify the speedup obtained by running the SU2 software on a parallelizable cluster and establish its effect on the solution time.

Chapter 2

Methodology

The methodology of this project was constructed based on the objectives and the tools were also based on them. As the main objective of the project was to validate SU2 as a CFD tool for the required flow problem, a geometry and mesh needed to be obtained or generated. As the airoils had already been designed and tested using other tools, the airfoil segment files were obtained from the AVATAR consortium. These segment files contained the airfoil geometries which needed to be validated. The sketches of these airfoils are shown in section 2.1.1.

The geometries generated were input into Pointwise, a mesh generation software, which has capability to output meshes into the native SU^2 format. Different mesh types such as structured, unstructured and hybrid were tested and their influence on the solution was assessed. A mesh independence study was also performed to make sure that the mesh size did not affect the solution obtained.

The mesh obtained from Pointwise was used to solve the flow problem in the SU^2 solver. The different parameters used and their effect on the solution are explained later.

A polar simulation was performed to simulate the flow around the airfoils for various angles of attacks. Only positive angles were considered to save time and computational resources. The results obtained were compared with the results of AVATAR partners and also with XFOIL[3] and RFOIL [4], an aerodynamic design tool based on XFOIL developed by ECN. Recently, the accuracy of RFOIL has been improved for thick airfoils.

The flow chart of the different steps is given below:

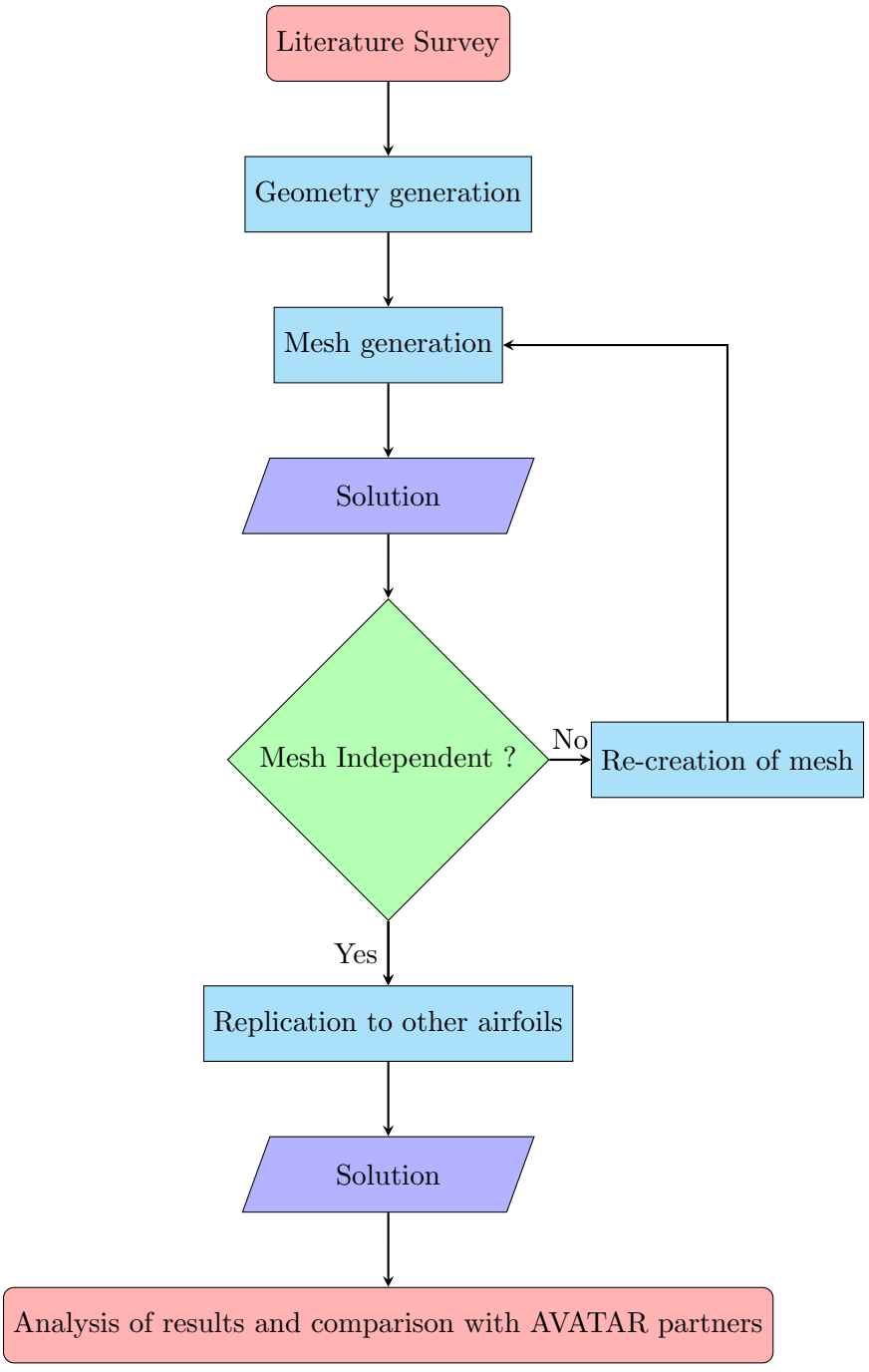


FIGURE 2.1: Methodology Flow chart

A summary of the tools used for the different steps for the project is given below:

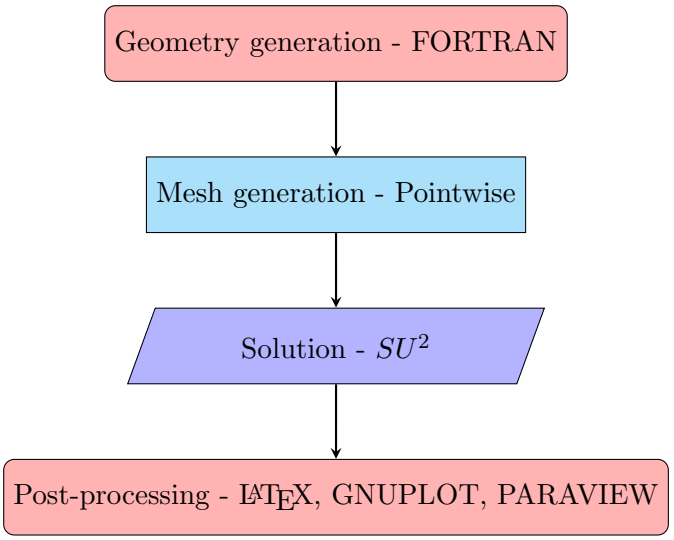


FIGURE 2.2: Summary of tools used

2.1 Pre-processing

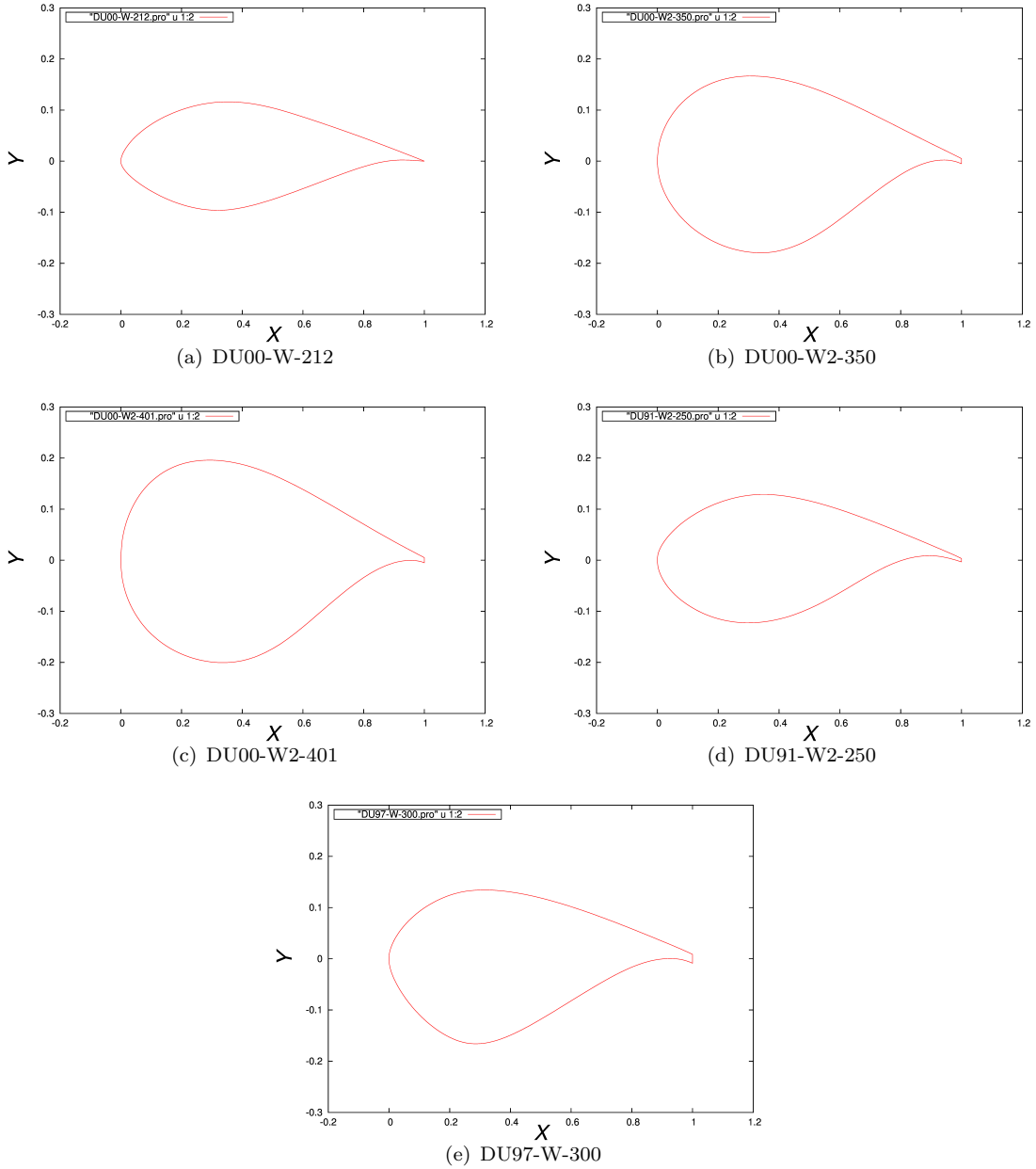
Pre-processing consists of generation of the airfoil geometry and the generation of mesh thereof. It basically consists of all the steps that are performed before the solution is computed.

2.1.1 Airfoil Geometries

The airfoils that were considered were obtained from the AVATAR consortium. Five airfoils were considered:

1. DU00-W-212 : Maximum thickness is 21.2 % of the chord length.
2. DU00-W2-350 : Maximum thickness is 35.0 % of the chord length.
3. DU00-W2-401 : Maximum thickness is 40.1 % of the chord length.
4. DU91-W2-250 : Maximum thickness is 25.0 % of the chord length.
5. DU97-W-300 : Maximum thickness is 30.0 % of the chord length.

The profiles of the airfoils that were considered are shown below:



(a) DU00-W-212

(b) DU00-W2-350

(c) DU00-W2-401

(d) DU91-W2-250

(e) DU97-W-300

FIGURE 2.3: Airfoil profiles that were used for comparisons

2.1.2 Mesh generation

The airfoil geometries shown in Fig 2.3 were converted to segment files and imported to Pointwise. The FORTRAN code used can be seen in Appendix A. Three different types of meshes exist:

1. Unstructured: No definite pattern, but a tessellation of the euclidean plane with triangular or tetrahedral elements.
2. Structured: Follows a definite pattern and has regular connectivity.
3. Hybrid: Contains portions having a definite pattern and portions that have no definite pattern.

There are advantages to each type of mesh. For example, the structured mesh, due to the definite pattern, when used to solve a flow problem needs lesser memory due to the fact that the data is structured and the structure is similar throughout the mesh. In contrast, for an unstructured mesh, more memory is required because there is no definite pattern and each node and element connectivity is required to fully describe the mesh. The structured meshes are also in general more accurate because the regularity in the mesh will result in cancellation of error terms in the interpolation variables which is not the case for unstructured meshes.

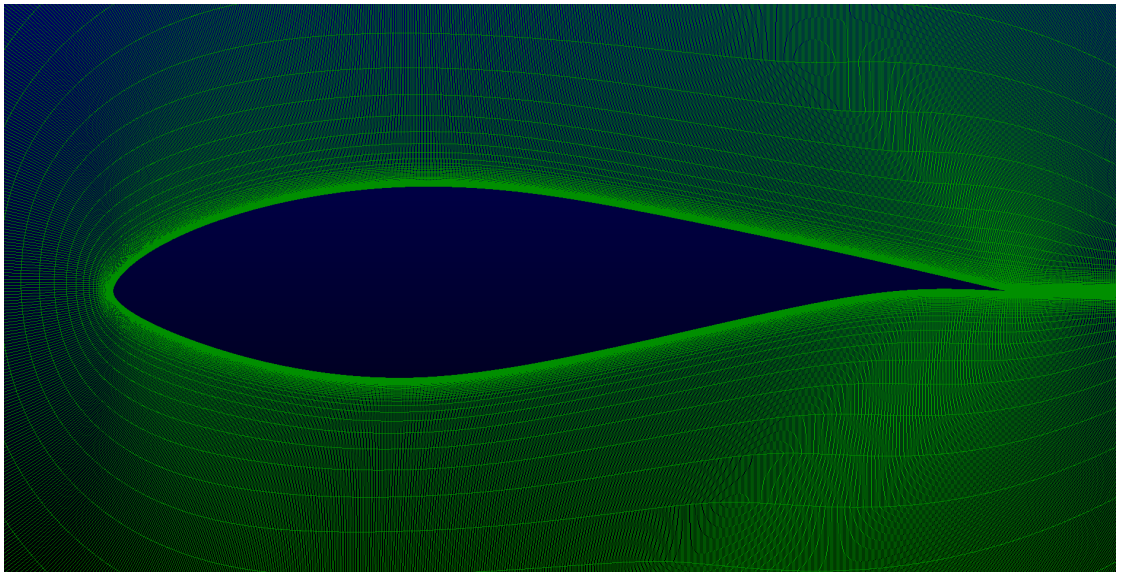
For a complicated geometry, the structured mesh becomes more expensive because to capture the details of the geometry accurately and preserve its properties, it becomes necessary to increase the number of elements. This is not the case for unstructured meshes.

Hybrid meshes are generally used when there is a refinement required in a certain region and memory saving is also required. It is advantageous in many situations where the geometry may be complex and refinement around the geometry is required.

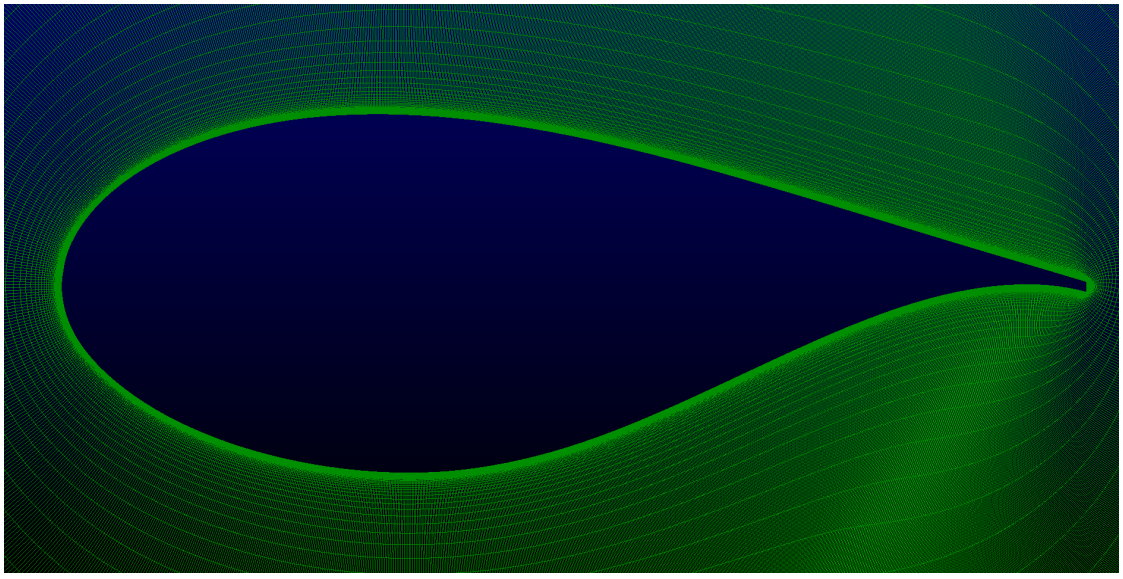
For our case, only the structured meshes have been considered, as the airfoil does not contain any complicated geometries and can be satisfactorily meshed with the structured meshing technique. One of the aspects that was considered in mesh generation was the initial Δs value at the airfoil wall. The initial Δs value, the height of the control volumes adjacent to the boundary and orthogonal to the airfoil wall, needs to be defined such that it can capture the flow above the airfoil. For our case, the initial Δs was set such that the y^+ is below 2 all along the airfoil wall. The y^+ defines the dimensionless wall distance in the flow and hence it is important that the y^+ is accurately captured in the flow. The y^+ was also checked after the completion of the simulation to ensure that the

it was indeed as it was set. The initial Δs and the growth rate defines the mesh density and number of cells in the mesh.

The meshes for various airfoils are as shown in 2.4 and 2.5. The meshes used for the mesh independence study are given later.

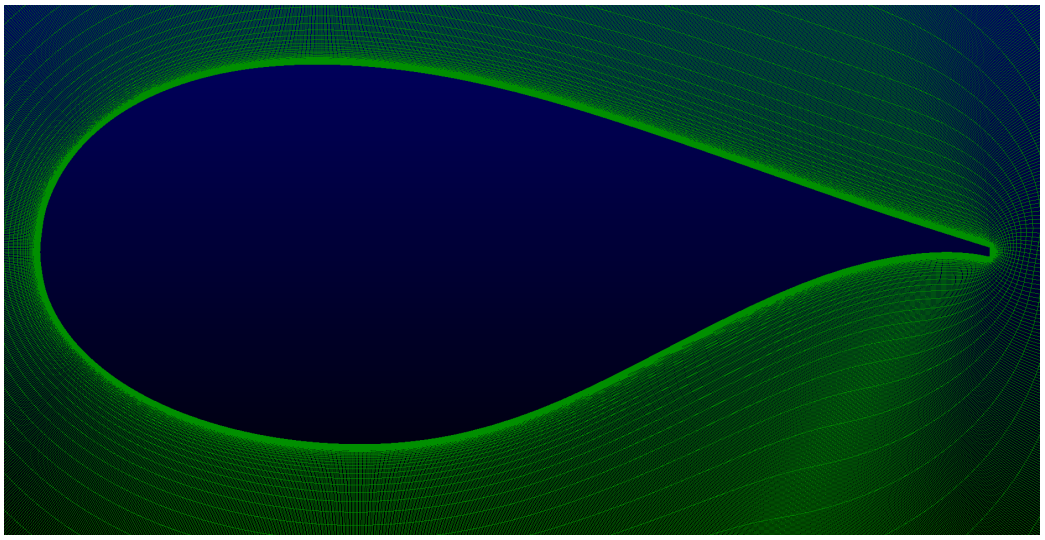


(a) DU00W212

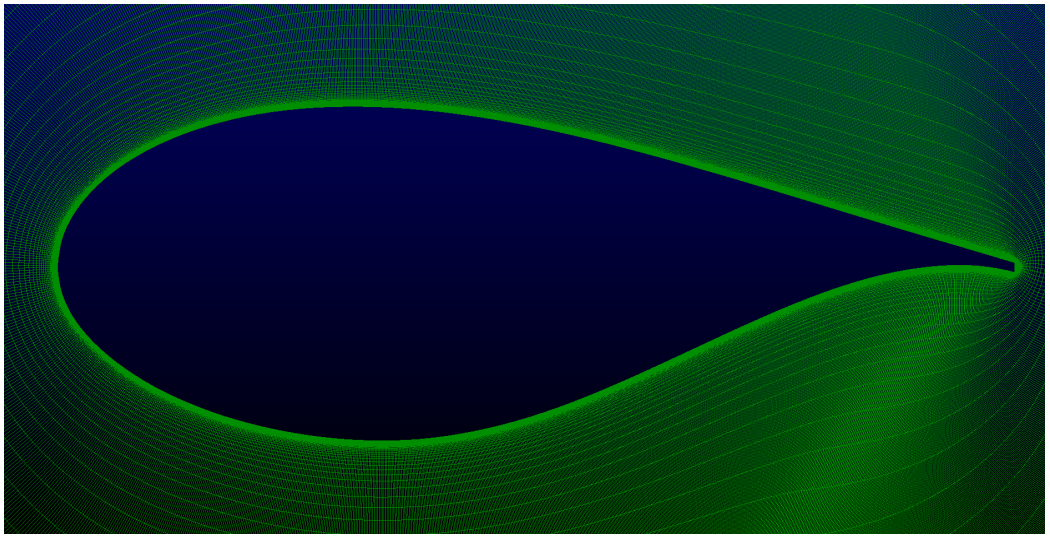


(b) DU00W2350

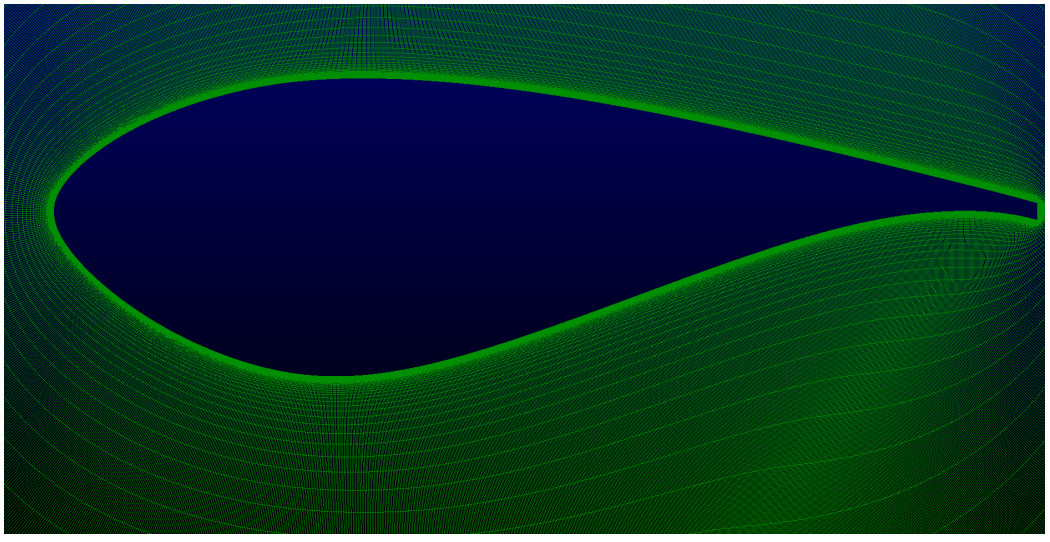
FIGURE 2.4: An overview of the meshes generated for various airfoil sections



(a) DU00W2401



(b) DU91-W2-250



(c) DU97-W-300

FIGURE 2.5: An overview of the meshes generated for various airfoil sections

2.2 Solution methods and strategies

Depending on the already available data which had to be validated, the airfoils given above were simulated at a range of Reynolds numbers. Table 2.1 shows the Reynolds numbers for the different airfoils .

TABLE 2.1: Reynolds numbers for different airfoils

Airfoil	Re
DU00-W-212	1.30E+007
	1.60E+007
	2.00E+007
DU00-W2-250	1.30E+007
	1.60E+007
	2.00E+007
DU00-W2-350	1.40E+007
DU91-W2-401	1.10E+007
DU97-W-300	1.70E+007

The different configurations that were chosen for the simulations are summarized in the Table 2.2

TABLE 2.2: Configurations for the SU^2 solver.

Parameter	Choice
Solver	Incompressible; Steady
Governing Equation	Navier Stokes
Turbulence model	Menter SST[5]
Density	1.00
Reynolds Number	Re
Freestream Velocity	(cos(α) sin(α) 0); α - Angle of attack
Freestream Viscosity	$\frac{1}{Re}$
Linear Solver	FGMRES
Precondiitoner for linear solver	LU symmetric Gauss Seidel
Multigrid	NO
Convective Numerical Method	Flux difference splitting scheme by Roe[2, 6].
Slope Limiter	Venkatakrishnan
Convergence criterion(P and K.E)	8 orders of magnitude reduction

The SU^2 solver is basically a compressible solver which can also solve incompressible flow problems using the artificial incompressibility method which is based on Chorin[7] and has been explained in detail in [2]. SU^2 uses the RANS (Reynolds Averaged Navier

Stokes) equations and a closure model to model the turbulent fluctuations. The Menter SST turbulence model was used because it gave a better accuracy than the SA turbulence model for this particular flow problem. The RANS equations take a time average of the mean flow properties. This is suitable for a steady flow. But for an unsteady case, the time averages must be taken at each time step and this may be expensive depending on the problem. For an unsteady flow it is necessary to use the Unsteady RANS (URANS) modelling. Turbulence modelling forms a very important aspect in this case and selection of a suitable turbulence model affects the accuracy of the solution obtained. The accuracy of RANS with respect to LES (Large eddy simulation) or DNS (Direct Numerical Simulation) may be lower but it requires a much less mesh resolution than required for LES and DNS.

Unfortunately in the latest version of SU^2, version 4.01, a transition model is not available and hence the simulations are performed for a fully turbulent flow. SU^2 is based on the discretization of the equations in non-dimensional form [2]. Hence we also have made the density and the velocity magnitude equal to unity and controlled the Reynolds number by making it equal to the inverse of the freestream viscosity. This makes it easy to input the various angles of attacks and the Reynolds number as required. To make sure that the solution obtained is converged, a convergence criteria of reduction of order 8 was used.

Boundary conditions are a very important aspect in a CFD method. SU^2 for the incompressible regime allows the far-field and the airfoil wall boundary conditions to be specified. The airfoil wall has a no-slip condition and the far-field has to be set such that a quiescent condition is maintained at the far-field walls. Here, this has been been ensured by taking the distance of the far-field at-least 70 chord lengths from the airfoil wall.

To obtain a complete polar plot, the simulation was run for angles of attack ranging from 0 to 24 degrees with an increment of 2 degrees. To make the solution converge faster and make the solutions for each of the angles of attack more accurate, the *solution of the previous angle of attack was used as the input for the next angle of attack.* A bash script was used to run the polar simulation. The script can be found in Appendix A.

The simulations were performed at the ECN HPC cluster, Reynolds, where each node of the cluster had the configuration given in Table 2.3.

The total number of cores available were $13 \times (12 \times 2) + 2 \times (18 \times 2) = 384$.

TABLE 2.3: System configuration of each of the nodes in the HPC cluster, Reynolds

Configuration	Value
Model Name:	Intel(R) Xeon(R) CPU E5-2670 @2.30GHz
Threads per core	2
Cores per socket	12 (or 18)
Sockets	2
NUMA nodes	2
L1d Cache	32K
L1i Cache	32K
L2 Cache	256K
L3 Cache	30720K

2.3 Post-processing

The solution obtained was processed for the Coefficient of Lift, C_l and Coefficient of Drag, C_d. The values obtained were compared with the results obtained by the AVATAR partners and also with XFOIL and RFOIL

The main observations were made with respect to the slope of the lift and drag curves in the linear region, the point of stall and the values of lift and drag coefficients after stall.

Chapter 3

Test Cases

The test cases were used to demonstrate the capabilities of SU^2 and verify its results on flow problems for which results are known very well. Two specific test cases were used here for two separate purposes. First was the NACA 0012 airfoil, which is a symmetric 12% thick NACA 4 digit airfoil used as a test case for most of the aerodynamic solvers. The second test case used was the DU00-W-212 airfoil which was used for a mesh independency study and in general for checking the mesh quality. Different mesh types and mesh sizes were simulated and the most efficient type and size was chosen for all the AVATAR cases.

3.1 NACA 0012 airfoil

The NACA 0012 airfoil is one of the basic test cases that is used to validate CFD methods because the experimental results are very well known and have been validated by many CFD solvers as well.The NACA 0012 airfoil which is a symmetric airfoil and the maximum thickness is 12% of the chord length.

The main objective of this test case as mentioned earlier was to validate SU^2 as a CFD solver for airfoils. The results obtained from SU^2 have been verified with experimental results, XFOIL and RFOIL. The C_d and C_l values have been plotted for various angles of attack from 0 to 20.

3.1.1 Geometry and mesh

The geometry of the NACA 0012 airfoil is as shown in Figure 3.1 and the mesh that has been used for the simulation is shown in Figure 3.2. The geometry was obtained from

XFOIL geometry generation tool and the mesh was generated using Pointwise. The mesh is a simple normal extrusion with a initial Δs value of 2.2×10^{-6}, which is called an O - grid.

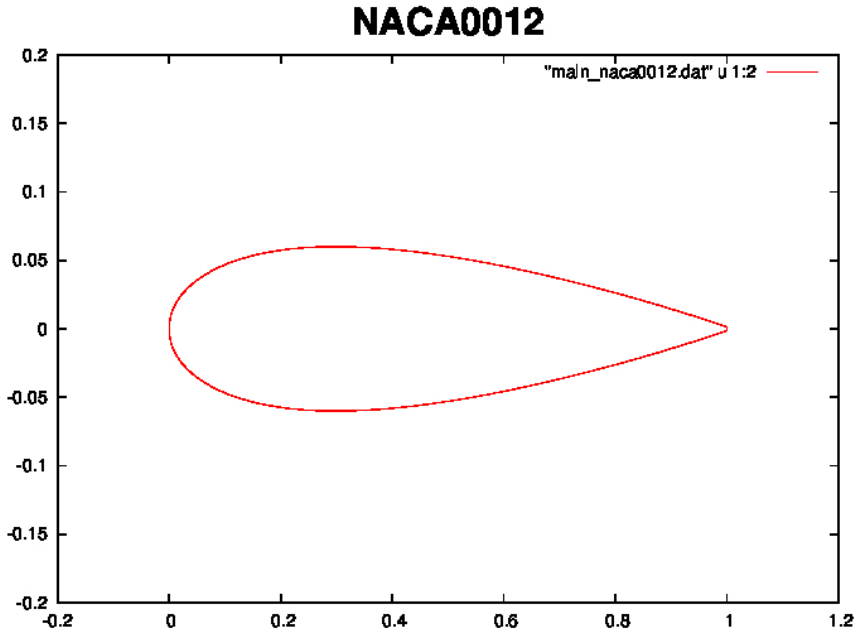


FIGURE 3.1: NACA 0012 geometry

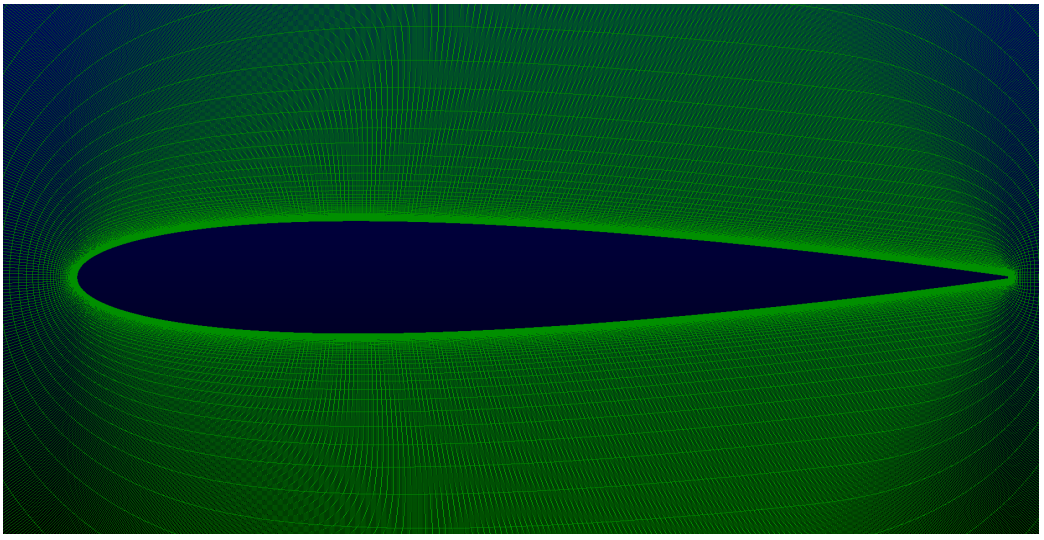


FIGURE 3.2: NACA 0012 mesh

3.1.2 Results and validation

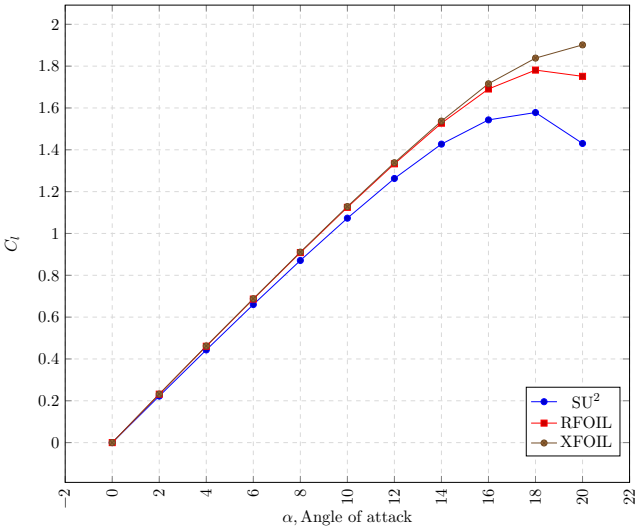
The configuration used for the NACA test case is shown in Table 3.1.

TABLE 3.1: Configurations for the NACA test case.

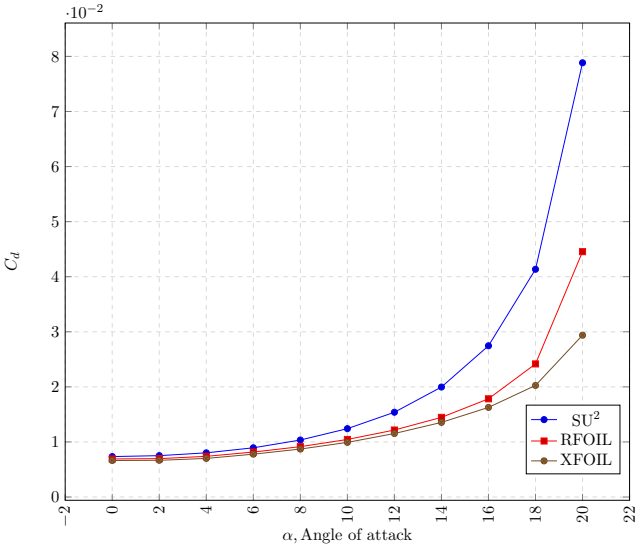
Parameter	Choice
Solver	Incompressible; Steady
Governing Equation	Navier Stokes
Turbulence model	Menter SST[5]
Density	1.00
Reynolds Number	3×10^6
Freestream Velocity	$(\cos(\alpha)\ \sin(\alpha)\ 0)$
Freestream Viscosity	$\frac{1}{Re}$
Linear Solver	FGMRES
Precondiitoner for linear solver	LU symmetric Gauss Seidel
Multigrid	NO
Convective Numerical Method	Flux difference splitting scheme by Roe[2, 6].
Slope Limiter	Venkatakrishnan
Convergence criterion(P and K.E)	8 orders of magnitude reduction

As observed from Figure 3.3, which shows that at a high Reynolds number the results from the various tools vary due to the different methodologies used. RFOIL and XFOIL use viscous-inviscid interaction together with tailor made closure relations whereas SU^2 uses CFD models and a turbulence model (Menter SST here) to close the equations. In the linear region, before stall for the airfoil, all the results are quite similar. As the stall region approaches the CFD tool SU^2 under-predicts lift and over-predicts drag compared to RFOIL and XFOIL.

A comparison with the experimental results is also given in the Figure 3.4 only for a Reynolds number of 3×10^6 as the experimental results for higher reynolds numbers were not available. As observed in the graphs, the stall is predicted by the SU^2 solver at about the same angle as observed in the experimental results. The C_d curve shows good agreement with its experimental counterpart. The experimental results are obtained from [8].

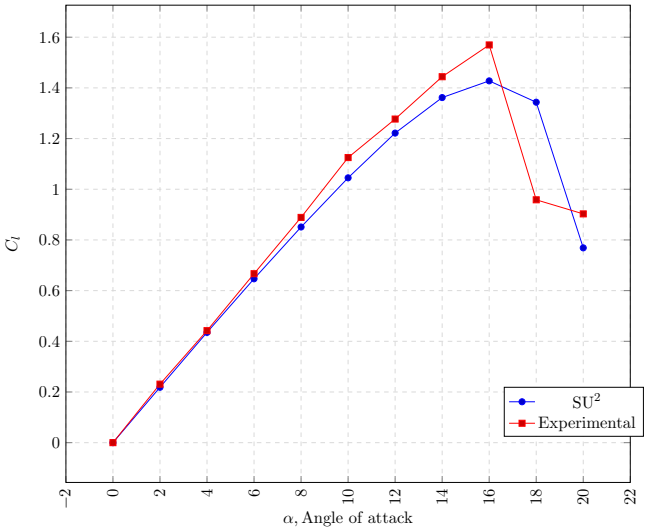


(a) NACA0012 at Re $= 13 \times 10^6$

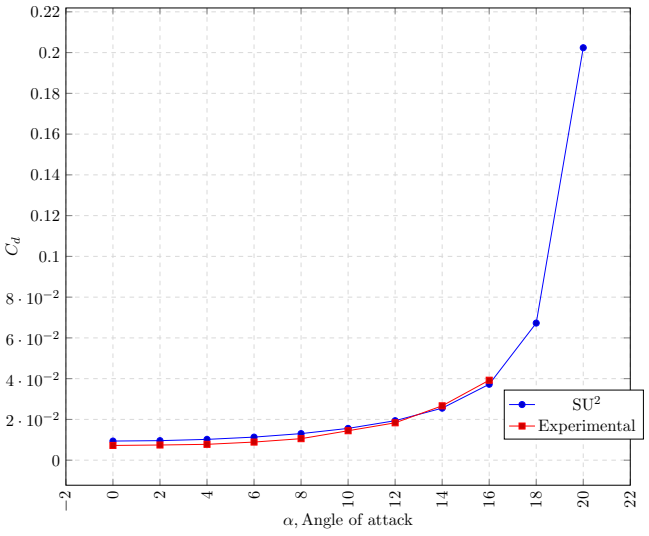


(b) NACA0012 at Re $= 13 \times 10^6$

FIGURE 3.3: C_l and C_d characteristics of NACA 0012 airfoil at Re $= 13 \times 10^6$

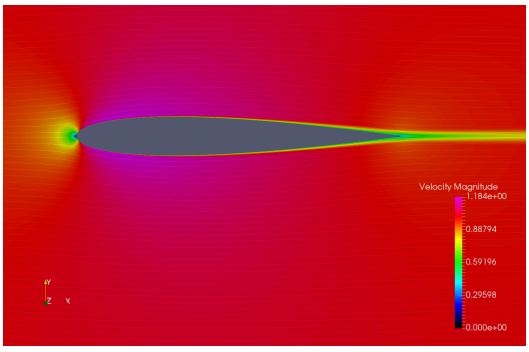


(a) C_l: NACA0012 at Re $= 3 \times 10^6$

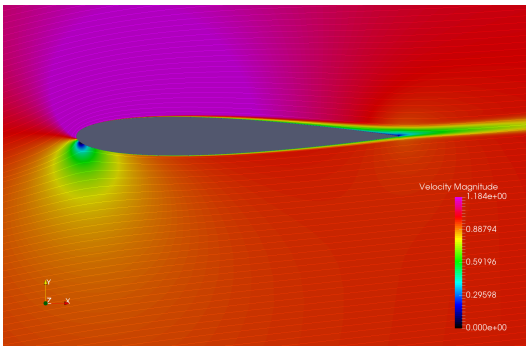


(b) C_d: NACA0012 at Re $= 3 \times 10^6$

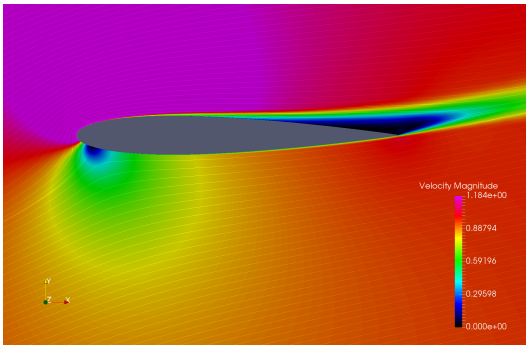
FIGURE 3.4: C_l and C_d characteristics of NACA 0012 airfoil at Re $= 3 \times 10^6$



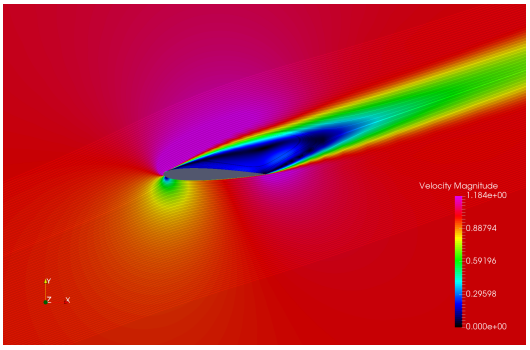
(a) NACA0012, $\alpha = 0$, $\mathrm{Re} = 3 \times 10^6$



(b) NACA0012, $\alpha = 8$, $\mathrm{Re} = 3 \times 10^6$



(c) NACA0012, $\alpha = 16$, $\mathrm{Re} = 3 \times 10^6$



(d) NACA0012, $\alpha = 20$, $\mathrm{Re} = 3 \times 10^6$

FIGURE 3.5: Velocity Magnitude contours with streamlines: NACA0012

3.2 DU-00-W-212 airfoil

The main objective of this test case as mentioned earlier was to study the mesh dependence on the solution and also validate the results with experimental values for high Reynolds numbers. The mesh independence study is based on a reynolds number of 13 million whereas the validation for high Re is based on a Reynolds number of 15 million. The C_d and C_l values have been plotted for various angles of attack from 0 to 20.

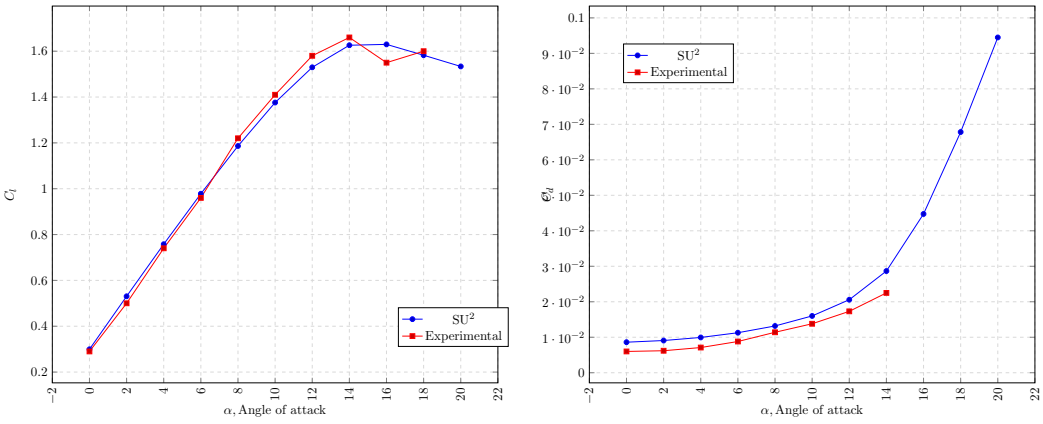
3.2.1 High Reynolds number validation

The AVATAR consortium has made available the experiments for the C_d and the C_l values for a Re of 15×10^6 and hence to validate SU^2, simulations were performed at a Re of 15×10^6 and the results are given in Figure 3.6. The aspects to be observed are:

1. The slope of the lift curve calculated by SU^2 is lower than the experimental values. This may be due to a fully turbulent simulation whereas the experimental data has only a turbulence intensity associated with it.

2. There is a clear over-prediction of drag. The behaviour of the two methods is similar. This over-prediction may be associated to the wake and unsteady effects in the flow present in the simulation, which can only be accurately captured by an unsteady simulation.

3. The experimental curve stalls around 14 degrees but SU^2 predicts the stall at 16 degrees at the given Reynolds number.



(a) C_l: Comparison of experimental and SU^2 results (b) C_d: Comparison of experimental and SU^2 results

FIGURE 3.6: High Reynolds number validation of DU-00-W-212 at Re $= 15 \times 10^6$

3.2.2 Mesh independence study

The objective of the mesh independence study is to study the solution for different mesh types and sizes and make sure that the solution does not change for the different meshes. A very coarse mesh would give results much faster but the accuracy may not be dependable as it is not able to capture all the features of the flow. On the other hand, a very fine mesh may capture all the features of the flow but may take a large amount of time which is not practical. The mesh independence study helps to determine the maximum size of the mesh (number of cells) that can be used practically without affecting the accuracy of the solution. For this reason, the mesh size is increased from the coarse to fine and the change in the solution parameters from one mesh size to the next is observed and when this change is below a certain acceptable criteria, it is said that the solution is independent of the mesh, which means that further refinement in the mesh will not bring any noticeable change in the solution (within the tolerance).

The criteria for the solution to be independent of the mesh was taken as about 0.1% change in the monitored variable under consideration between a given mesh and a refined one.

The types of meshes that were tested on were the following:

Structured: A boundary conforming grid consisting of quadrilaterals throughout the domain. It was made sure that the grid is orthogonal to the airfoil surface.

1. **C-grid**: The zero degree angle of attack wake line explicitly created to ensure lower skewness at the trailing edge.
2. **O-grid**: A traditional extrusion of the grid from the airfoil wall to make sure it conforms with the boundary.

The Table 3.2 shows the mesh characteristics for the different meshes used for the mesh independence study. The initial Δs was chosen such that the value of y^+ near the airfoil wall would be lesser than 2. After completion of the simulation it was verified that the y^+ was as required.

The Table 3.3 summarizes the simulation times required for various mesh sizes. As seen the very fine O grid takes impractical amount of time compared to the fine O grid giving results which are about 0.1 % change in solution in the linear region of the lift curve. The C grid also takes a lot of time due to a unnatural and artificial wake line created and an extrusion about it. Though this reduces the skewness in the cells near the trailing edge of the airfoil, it does not give a good convergence. On the other hand the O grid

TABLE 3.2: Mesh independence study characteristics

Sl. No.	Number of cells	Coarse or Fine	Type of mesh	Initial Δs
1	374750	Very fine	Structured-O Grid	1.1×10^{-6}
2	245754	Fine	Structured-O Grid	2.2×10^{-6}
3	117764	Coarse	Structured-O Grid	4.4×10^{-6}
4	292032	Fine	Structured-C Grid	2.2×10^{-6}
5	163064	Coarse	Structured-C Grid	4.4×10^{-6}

gives a much better convergence even though the skewness is relatively higher at the trailing edge of the airfoil.

TABLE 3.3: Mesh independence simulation timings in min

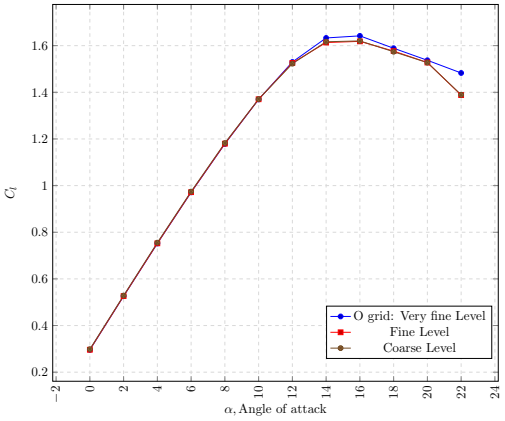
Mesh Type	Till $\alpha = 20$	Till $\alpha = 22$	Till $\alpha = 24$
Very fine O grid	2184	2513	3848
Fine O grid	795	924	1829
Coarse O grid	286	706	1129
Fine C grid	1903	2954	4005
Coarse C grid	2044	2644	3244

3.2.3 Results and validation

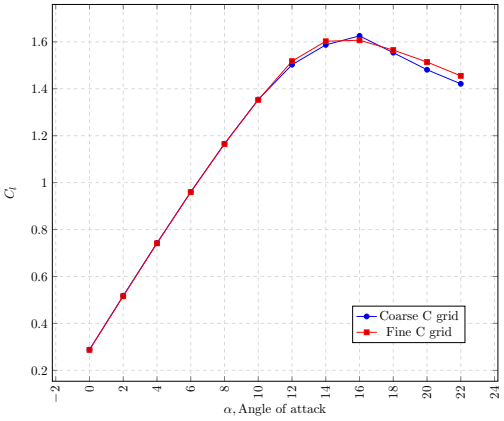
As the objective of the mesh independence study was to establish the mesh beyond which refinement would not affect the solution beyond a certain limit since in true essence mesh independence cannot be achieved. The error in the C_l and C_d values at each angle of attack has been shown for the different types of meshes in Tables 3.4 and 3.5.

The Figure 3.7 and Figure 3.8 show the lift and drag curves for the different meshes. As observed, there is a very slight difference between the very fine and fine grid levels of the O grid, where as the difference between the coarse and fine grid in both the C grid and the O grid is quite large. These errors have been tabulated in Table 3.4 and Table 3.5.

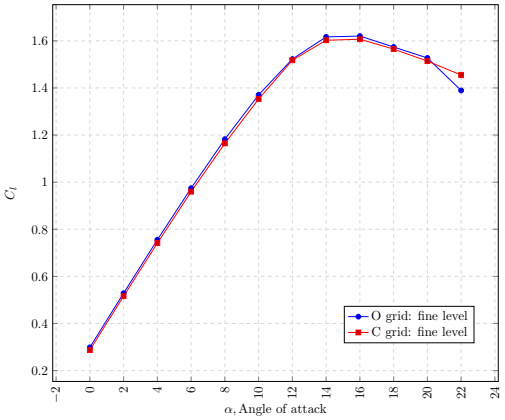
As observed, the change in the solution from the fine level of the O grid to the very fine level is very small(within 0.1 % in the linear, and with 7% in the other regions). Also, the time taken for the very fine grid level is much higher than that of the fine level. Hence it was decided that the fine level (approximately 250000 cells, refer Table 3.2) would be used for all the airfoils which corresponds to an initial Δs value of 2.2×10^{-6}.



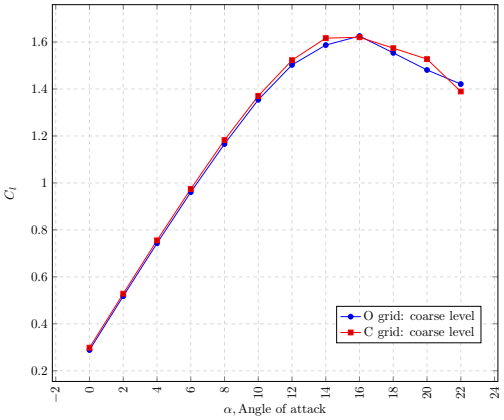
(a) C_l: Comparison of O grid at very fine, fine level and coarse levels



(b) C_l: Comparison of C grid at fine and coarse levels



(c) C_l: Comparison of O and C grids at fine level

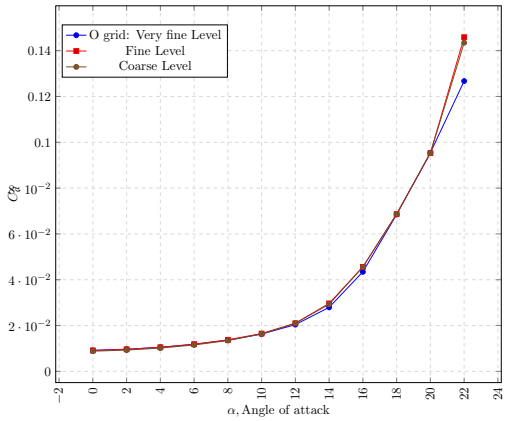


(d) C_l: Comparison of O and C grids at coarse level

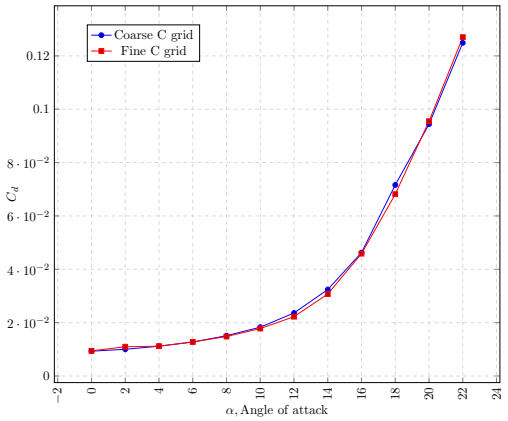
FIGURE 3.7: C_l: Mesh independence study of DU-00-W-212 at Re $= 13 \times 10^6$

TABLE 3.4: Mesh independence study, error in percentage for C grid

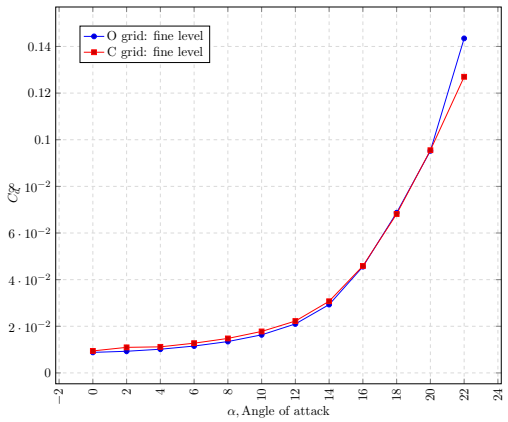
Angle of attack	C_l: Coarse to fine	C_d: Coarse to fine
0	0.37	-1.46
2	0.37	-8.70
4	0.22	-0.19
6	0.11	-0.07
8	0.10	2.17
10	0.08	2.88
12	-1.00	6.20
14	-0.97	5.48
16	1.16	0.88
18	-0.71	5.09
20	-2.18	-1.16
22	-2.33	-1.66



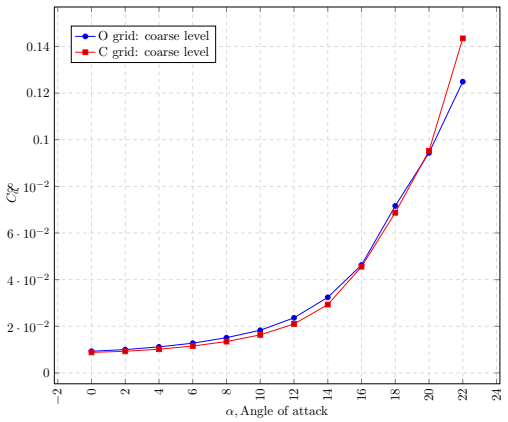
(a) C_d: Comparison of O grid at very fine, fine level and coarse levels



(b) C_d: Comparison of C grid at fine and coarse levels



(c) C_d: Comparison of O and C grids at fine level



(d) C_d: Comparison of O and C grids at coarse level

FIGURE 3.8: C_d: Mesh independence study of DU-00-W-212 at Re $= 13 \times 10^6$

TABLE 3.5: Mesh independence study, error in percentage for O grid

Angle of attack	C_l:Coarse to fine	C_d:Coarse to fine	C_l:Fine to very fine	C_d: Fine to very fine
0	0.79	-3.71	-0.18	0.52
2	0.42	-3.61	-0.11	0.30
4	0.39	-3.38	-0.08	-0.09
6	0.25	-2.76	-0.09	-0.45
8	0.23	-2.29	-0.08	-0.94
10	0.03	-1.15	-0.06	-1.42
12	-0.20	-0.02	0.27	-2.99
14	0.21	-1.20	1.20	-6.14
16	0.12	-0.15	1.44	-5.01
18	-0.18	0.08	0.77	-0.25
20	-0.03	-0.04	0.62	0.17
22	0.09	-1.66	6.39	-15.11

Chapter 4

Results and Validation of AVATAR airfoils

The coefficient of drag and lift of different airfoils that were tested are shown. The results are first compared with XFOIl and RFOIL. Next they are compared with the AVATAR results obtained from the AVATAR consortium. Observations are drawn and conclusions are drawn based on the observations. Velocity contour plots are shown for some angles of attacks for all the airfoils.

4.1 Comparison of SU^2 with RFOIL and XFOIL

As explained before, XFOIL and RFOIL are tools that solve integral boundary layer equations together with clossure relations using the viscid-inviscid interaction schemes to model the flow around the airfoils. XFOIL was the original version and its details can be found in [3]. RFOIL was developed based on XFOIL but specifically to be adapted for rotating section of airfoils. Its details can be found in [4]. SU^2 on the other hand uses the Navier Stokes equations with a turbulence model (Menter SST here) to solve for the flow over the airfoil.

The airfoils were simulated for the Reynolds numbers as given in Table 2.1. The Figure 4.1 and Figure 4.2 show the C_l and C_d for the DU-00-W-212 and the DU-00-W2-250 airfoils. The following observations can be made:

1. There is a clear under-prediction of C_l in the lift curve and an over-prediction of C_d in the drag curve. Both these are observed in the nearly linear region of the lift and drag curves. This is attributed to the fact that SU^2 solves for a fully turbulent

flow and uses a turbulence model to close the equations. XFOIL and RFOIL on the other hand solve boundary layer equations near the airfoil wall and potential flow equations elsewhere. This difference in solution methods brings about a change in the results. Further their tools have a transition model implemented.

2. XFOIL predicts stall much later than SU^2 and RFOIL. As will be seen later the AVATAR results as well seem to agree with the results of SU^2 and RFOIL.

3. **For DU-00-W-212:** The general trend of the lift and drag curves is similar in both RFOIL and SU^2. RFOIL predicts stall earlier than SU^2 and after stall the slope of the lift curve tends to become constant in RFOIL, but the slope of the lift curve tends to decrease in SU^2.

4. **For DU-00-W2-250:** The general trend of the lift and drag curves are again similar.

5. **For DU-00-W2-250:** For higher angles of attack, RFOIL fails to converge due to separation of the flow and hence the corresponding C_d values are not available.

6. As the Reynolds number increases, the behaviour of RFOIL and SU^2 remain same but XFOIL varies. This confirms that for these airfoils, the results of RFOIL and SU^2 is more reliable than XFOIL.

7. After stall, in RFOIL, it is observed that the slope of the drag curve increases more than that of the corresponding curve in SU^2.

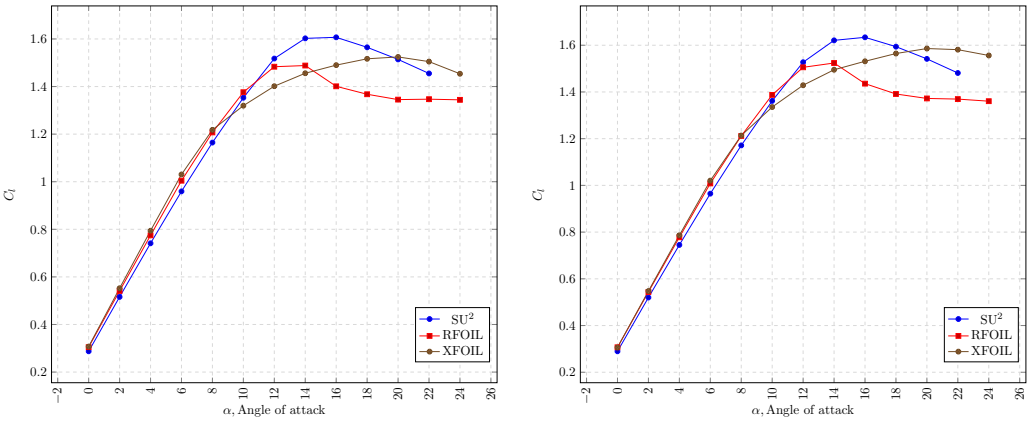
The following observations are made considering the other 3 airfoils (Figures 4.3 and 4.4):

1. **For DU-00-W2-350**: The lift curves of RFOIL and SU^2 are in good agreement except in two regions:

   (a) Zero angle of attack: RFOIL predicts stall at very low positive angles of attack but SU^2 does not. The kink observed in the RFOIL lift curve is due to this. SU^2 may also predict such a behaviour, but when simulated from negative angles to positive angles, that is for the whole spectrum of angle of attacks.

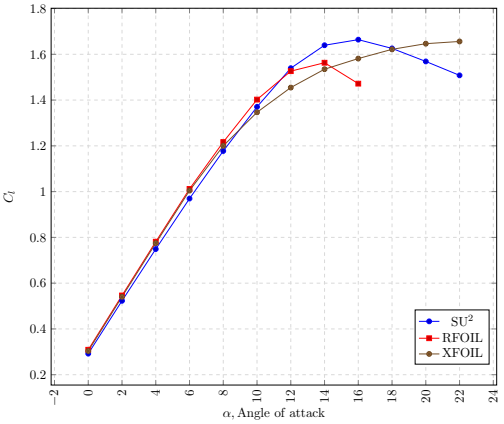
   (b) The airfoil is 35 % thick and hence the results from XFOIL and RFOIL are not very reliable after stall because of the presence of separation.

   The drag curves of RFOIL and SU^2 are in good agreement throughout the range with slight change in the slopes with particular increase in slope of the SU^2 lift curve after slope due to the separation of flow which has been known to increase

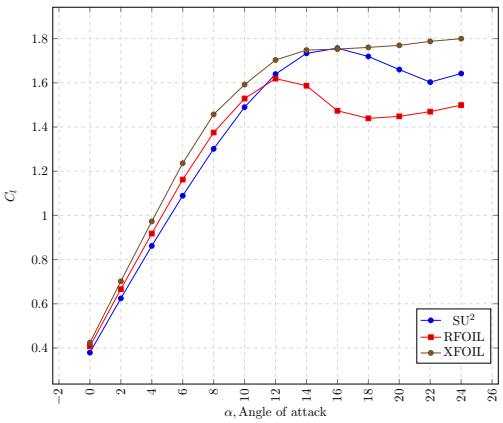


(a) DU-00-W-212 at Re $= 13 \times 10^6$

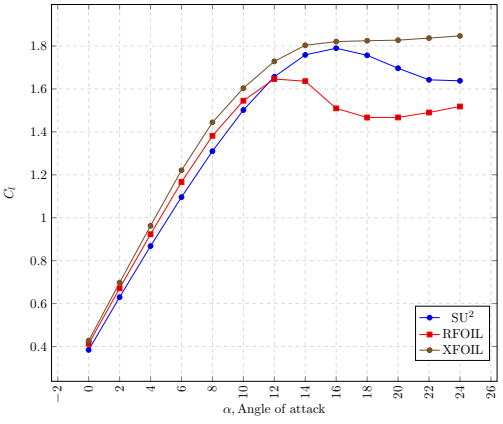
(b) DU-00-W-212 at Re $= 16 \times 10^6$



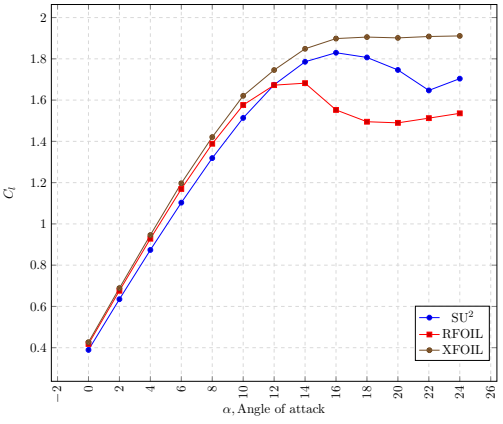
(c) DU-00-W-212 at Re $= 20 \times 10^6$



(d) DU-00-W2-250 at Re $= 13 \times 10^6$



(e) DU-00-W2-250 at Re $= 16 \times 10^6$

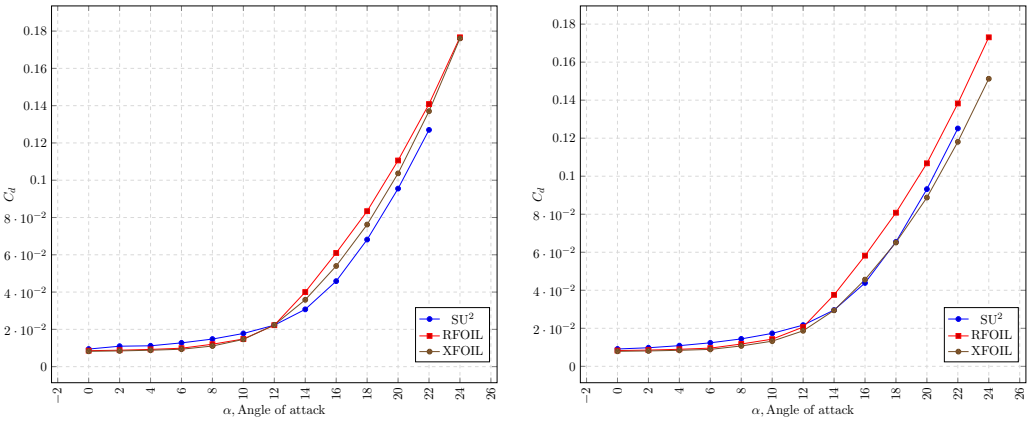


(f) DU-00-W2-250 at Re $= 20 \times 10^6$

FIGURE 4.1: C_l characteristics of airfoils: Part 1

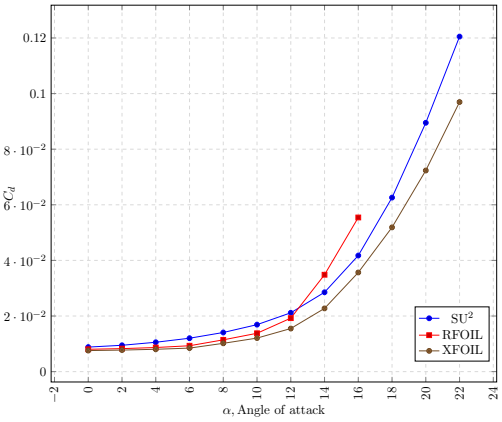
the drag considerably [9]. Though the reference is for low reynolds number cases, similar behaviour may be expected for higher reynolds number because the separation does not allow the pressure to be recovered and hence the drag increases.

2. **For DU-91-W2-401**: The lift curve obtained through RFOIL predicts a large drop in lift at low positive angle of attack which implies that the flow has separated.

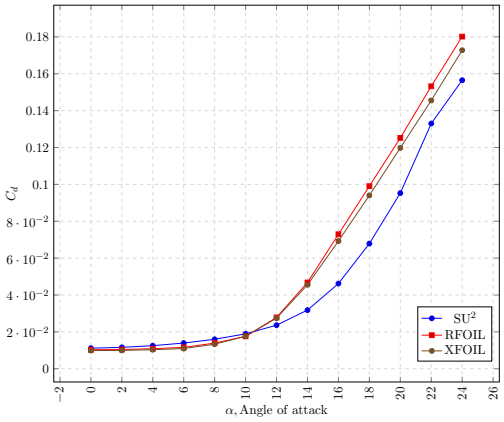


(a) DU-00-W-212 at Re $= 13 \times 10^6$

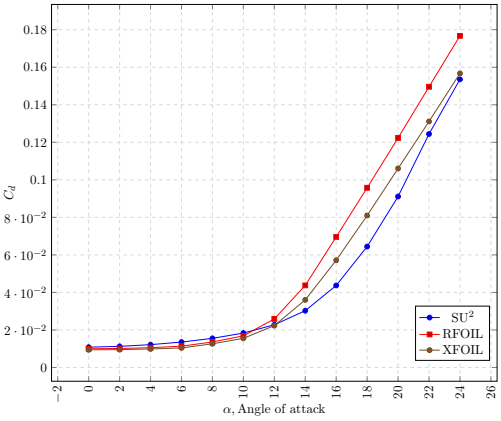
(b) DU-00-W-212 at Re $= 16 \times 10^6$



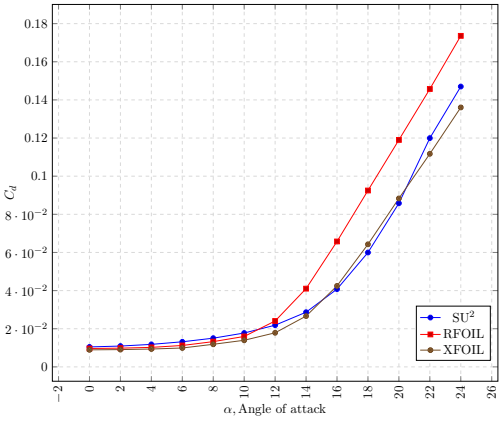
(c) DU-00-W-212 at Re $= 20 \times 10^6$



(d) DU-00-W2-250 at Re $= 13 \times 10^6$



(e) DU-00-W2-250 at Re $= 16 \times 10^6$



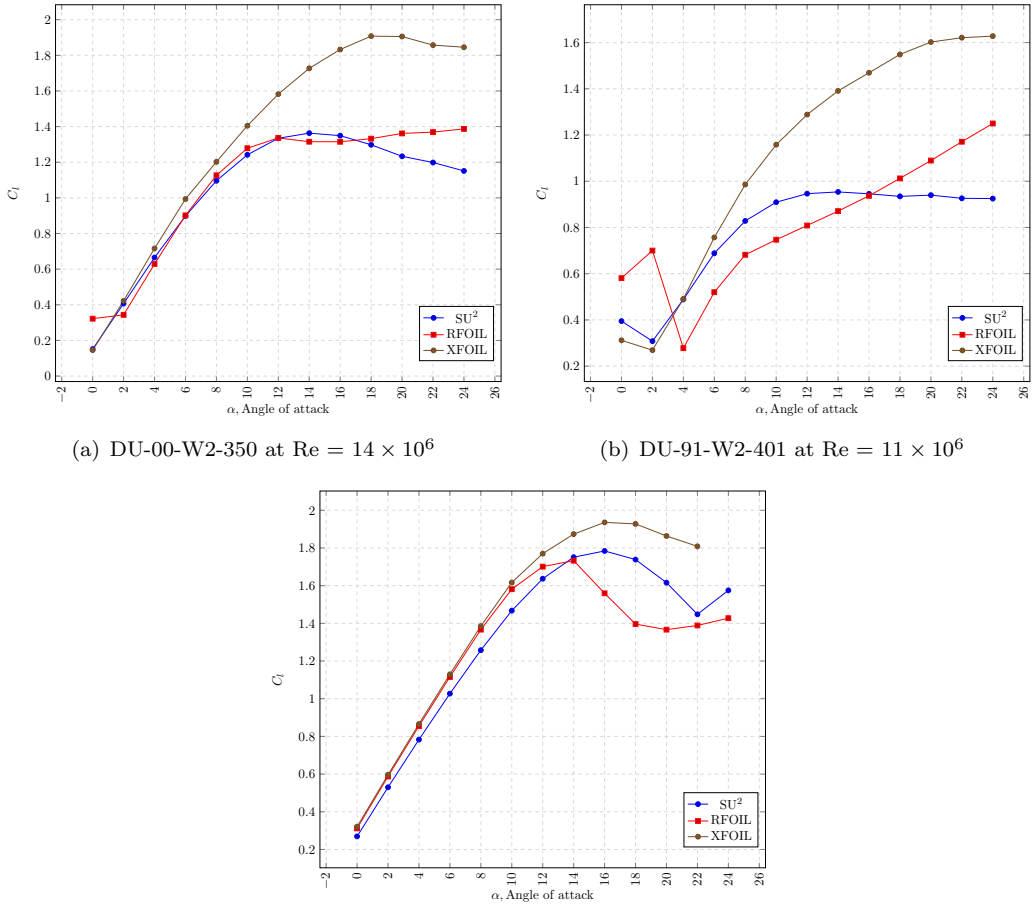
(f) DU-00-W2-250 at Re $= 20 \times 10^6$

FIGURE 4.2: C_d characteristics of airfoils: Part 1

But this is not the case as can be observed in Figure 4.10(a). It is observed that the flow indeed separates and then as the angle of attack increases, the flow reattaches on the suction side but the drop in lift as predicted through RFOIL would require a much larger separation. Also RFOIL predicts a continuous increase in C_l which is again not the case. This behavaiour from RFOIL may be attributed to the fact

that due to separation, RFOIL is not able to converge and give accurate results. SU^2 on the other hand being a CFD method, can predict the C_l and C_d better than RFOIL and XFOIL for very thick airfoils. The drag curve has the same characteristics and behaviour as for the other airfoils. One more aspect that can be observed is that C_d is over-predicted by RFOIL which was not the case for the other airfoils.

3. **For DU-97-W-300**: The lift curve behaviour for this airfoil is similar to that of the DU-00-W-212 and the DU-00-W2-250 airfoils. The only difference in drag curve is that after stall, there is a clear over-prediction of C_d through RFOIL but the behaviour is similar.

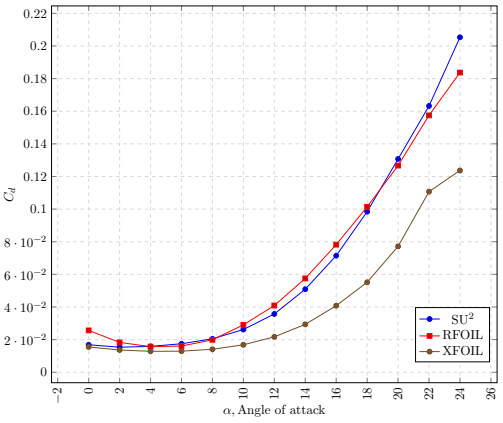


(a) DU-00-W2-350 at Re $= 14 \times 10^6$

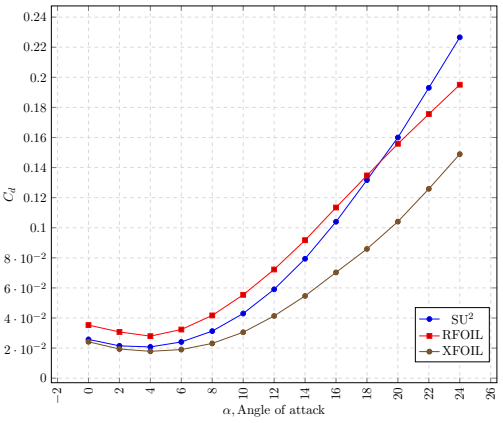
(b) DU-91-W2-401 at Re $= 11 \times 10^6$

(c) DU-97-W-300 at Re $= 17 \times 10^6$

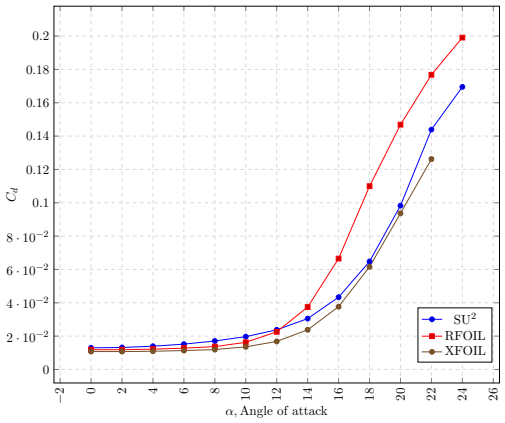
FIGURE 4.3: C_l characteristics of airfoils: Part 2



(a) DU-00-W2-350 at Re $= 14 \times 10^6$

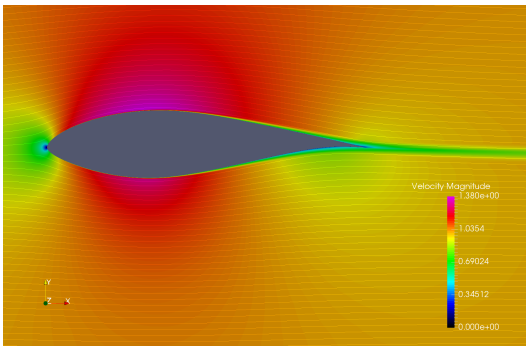


(b) DU-91-W2-401 at Re $= 11 \times 10^6$

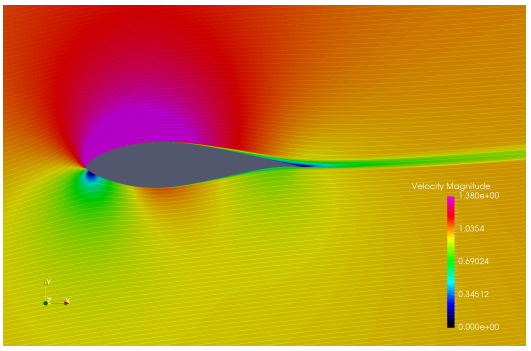


(c) DU-97-W-300 at Re $= 17 \times 10^6$

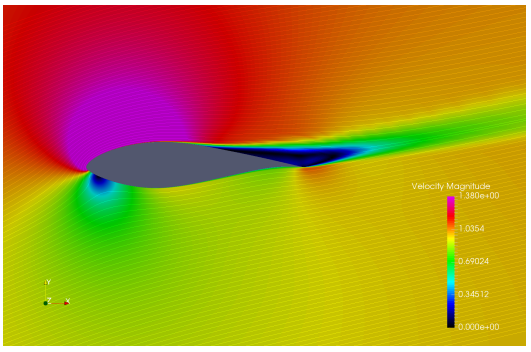
FIGURE 4.4: C_d characteristics of airfoils: Part 2



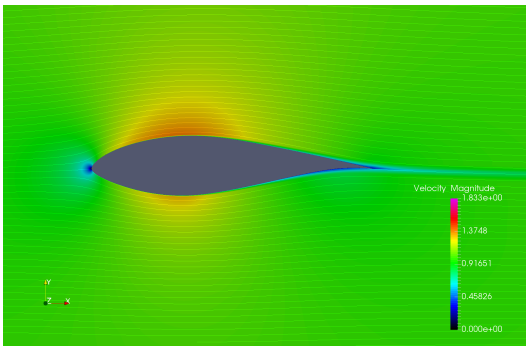
(a) DU00-W-212, $\alpha = 0$, $\mathrm{Re} = 13 \times 10^6$



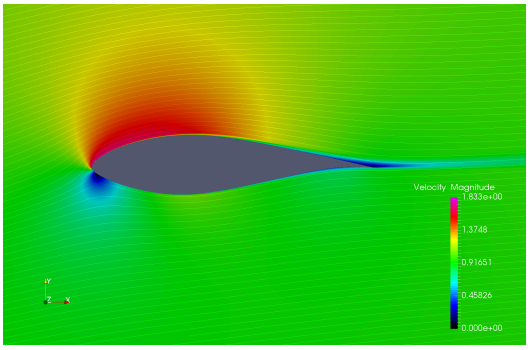
(b) DU00-W-212, $\alpha = 8$, $\mathrm{Re} = 13 \times 10^6$



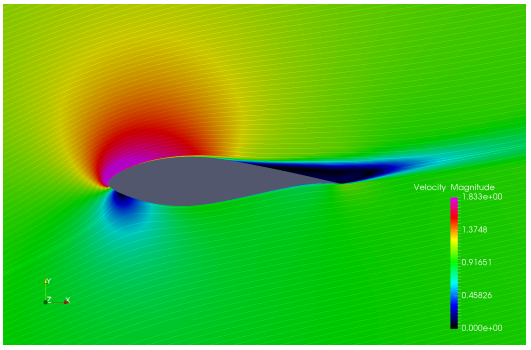
(c) DU00-W-212, $\alpha = 16$, $\mathrm{Re} = 13 \times 10^6$



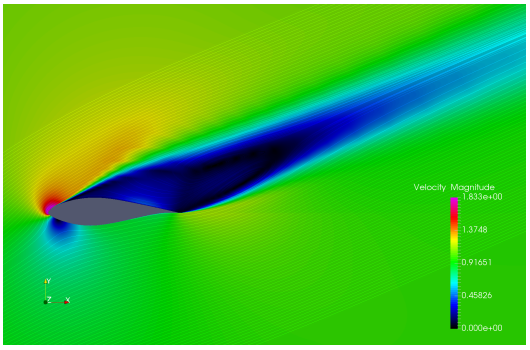
(d) DU00-W-212, $\alpha = 0$, $\mathrm{Re} = 16 \times 10^6$



(e) DU00-W-212, $\alpha = 8$, $\mathrm{Re} = 16 \times 10^6$

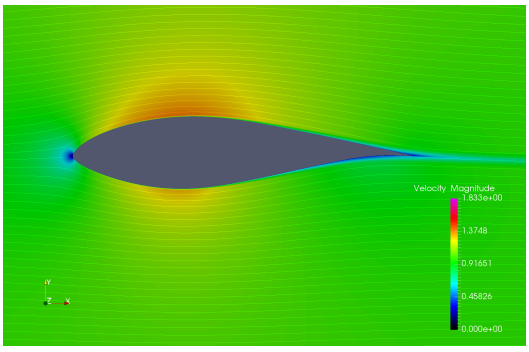


(f) DU00-W-212, $\alpha = 16$, $\mathrm{Re} = 16 \times 10^6$

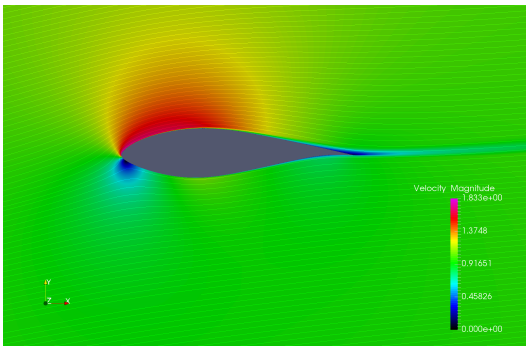


(g) DU00-W-212, $\alpha = 24$, $\mathrm{Re} = 16 \times 10^6$

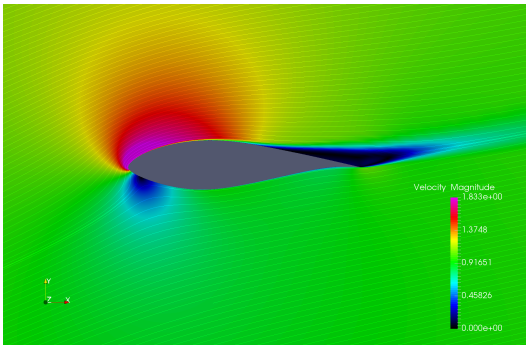
FIGURE 4.5: Velocity Magnitude contours with streamlines: DU00-W-212, Part 1



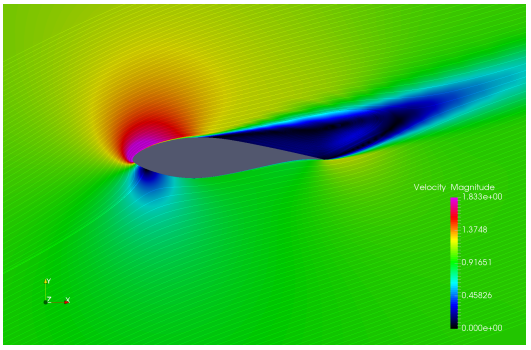
(a) DU00-W-212, $\alpha = 0$, Re $= 20 \times 10^6$



(b) DU00-W-212, $\alpha = 8$, Re $= 20 \times 10^6$

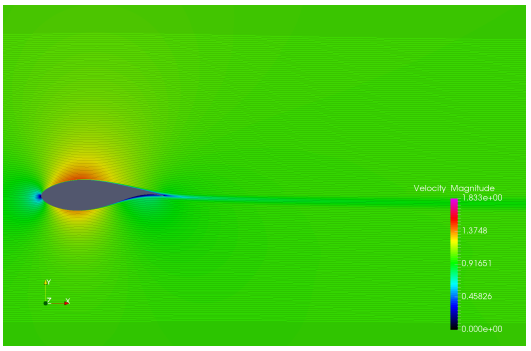


(c) DU00-W-212, $\alpha = 16$, Re $= 20 \times 10^6$

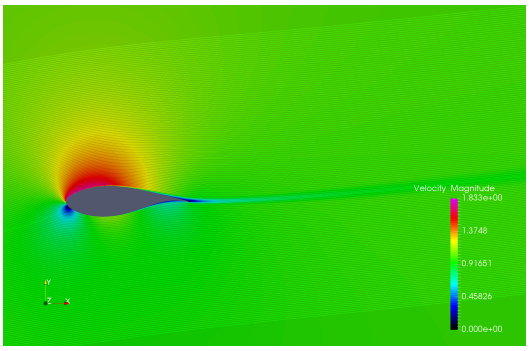


(d) DU00-W-212, $\alpha = 22$, Re $= 20 \times 10^6$

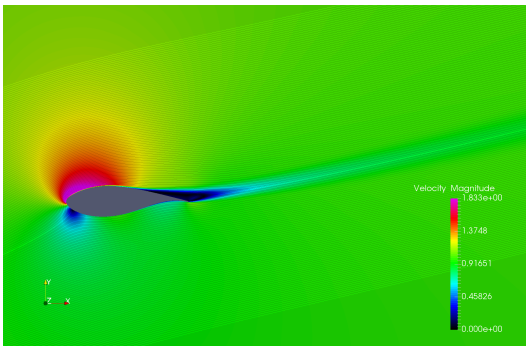
FIGURE 4.6: Velocity Magnitude contours with streamlines: DU00-W-212, Part 2



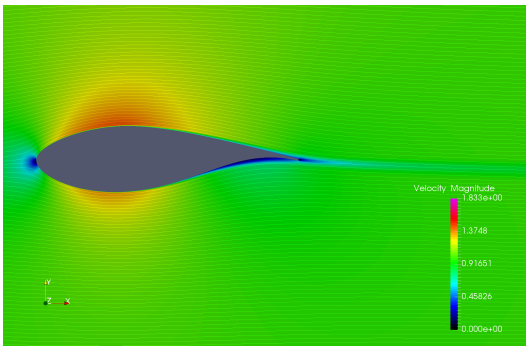
(a) DU00-W2-250, $\alpha = 0$, Re $= 13 \times 10^6$



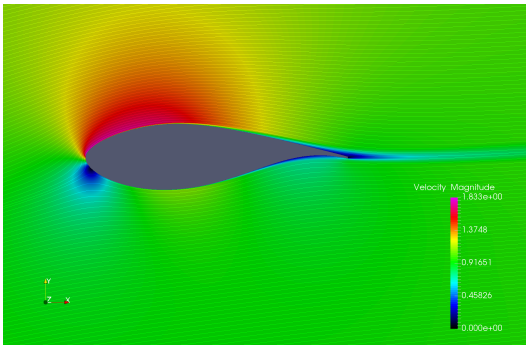
(b) DU00-W2-250, $\alpha = 8$, Re $= 13 \times 10^6$



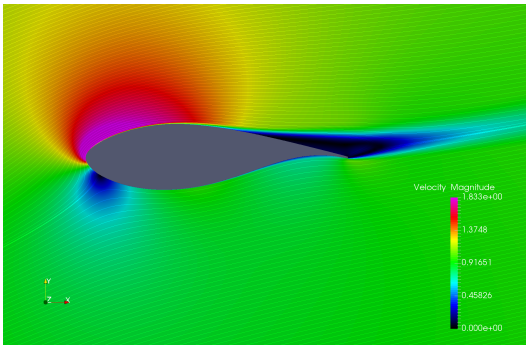
(c) DU00-W2-250, $\alpha = 16$, Re $= 13 \times 10^6$



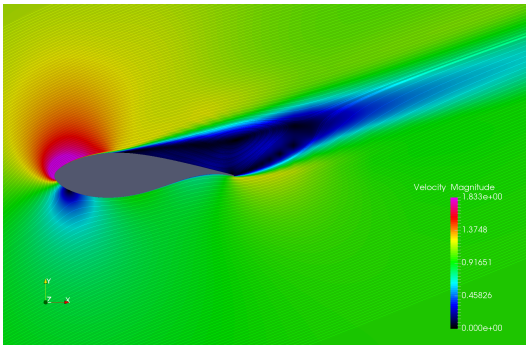
(d) DU00-W2-250, $\alpha = 0$, Re $= 16 \times 10^6$



(e) DU00-W2-250, $\alpha = 8$, Re $= 16 \times 10^6$

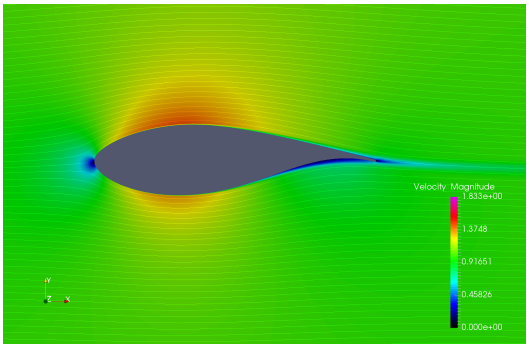


(f) DU00-W2-250, $\alpha = 16$, Re $= 16 \times 10^6$

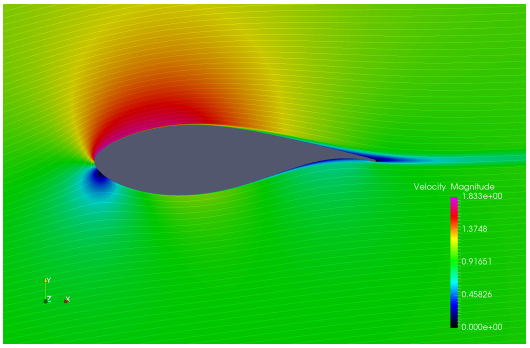


(g) DU00-W2-250, $\alpha = 24$, Re $= 16 \times 10^6$

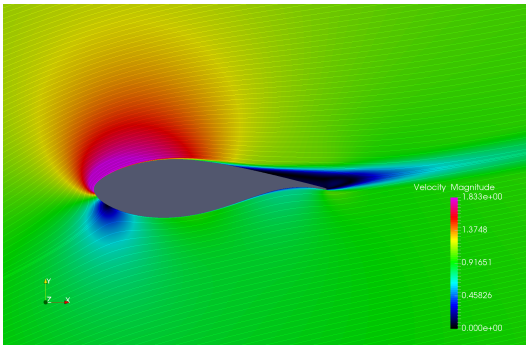
FIGURE 4.7: Velocity Magnitude contours with streamlines: DU00-W2-250, Part 1



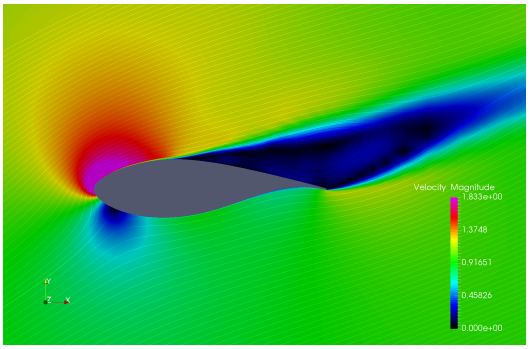
(a) DU00-W2-250, $\alpha = 0$, Re $= 20 \times 10^6$



(b) DU00-W2-250, $\alpha = 8$, Re $= 20 \times 10^6$

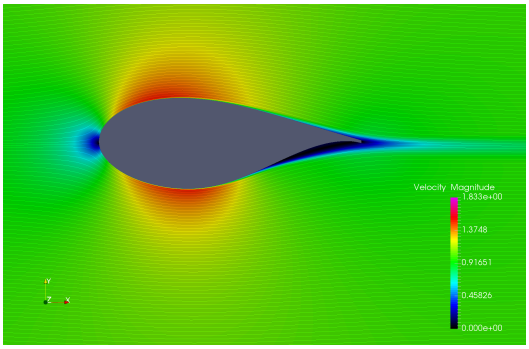


(c) DU00-W2-250, $\alpha = 16$, Re $= 20 \times 10^6$

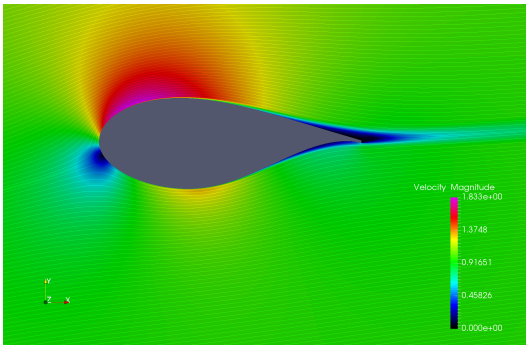


(d) DU00-W2-250, $\alpha = 24$, Re $= 20 \times 10^6$

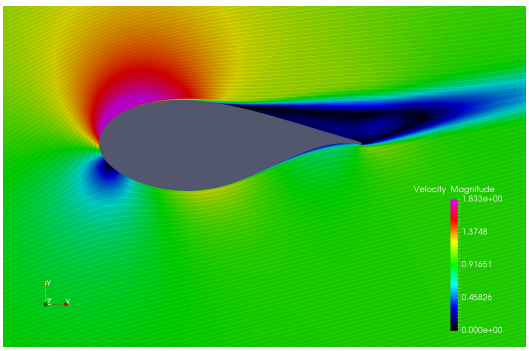
FIGURE 4.8: Velocity Magnitude contours with streamlines: DU00-W2-250, Part 2



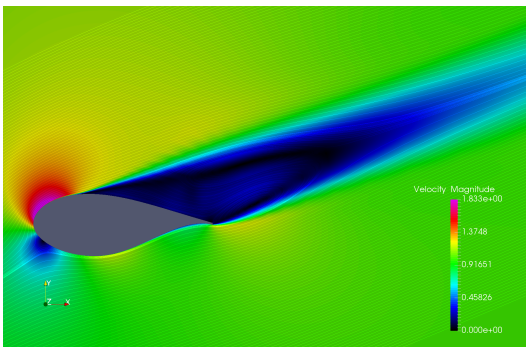
(a) DU00-W2-350, $\alpha = 0$, Re $= 14 \times 10^6$



(b) DU00-W2-350, $\alpha = 8$, Re $= 14 \times 10^6$

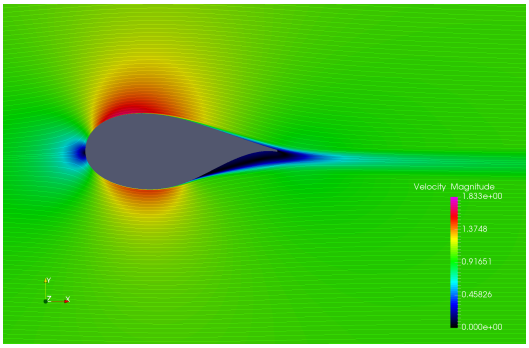


(c) DU00-W2-350, $\alpha = 16$, Re $= 14 \times 10^6$

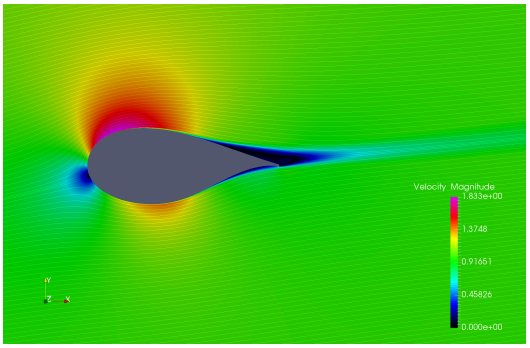


(d) DU00-W2-350, $\alpha = 24$, Re $= 14 \times 10^6$

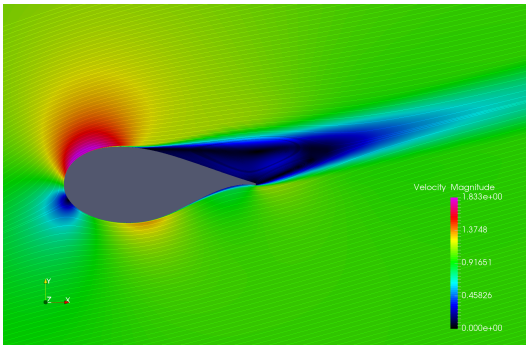
FIGURE 4.9: Velocity Magnitude contours with streamlines: DU00-W2-350



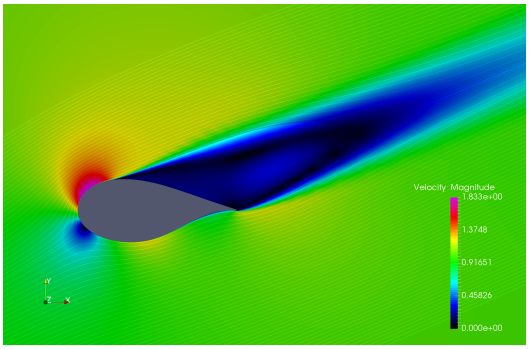
(a) DU91-W2-401, $\alpha = 0$, Re $= 11 \times 10^6$



(b) DU91-W2-401, $\alpha = 8$, Re $= 11 \times 10^6$

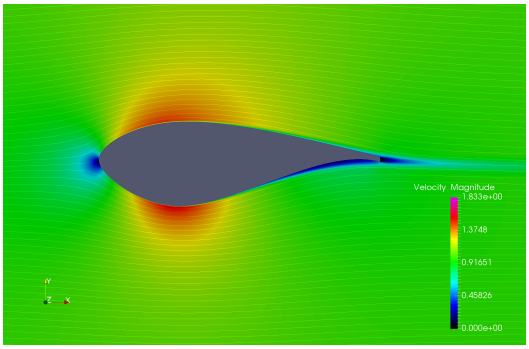


(c) DU91-W2-401, $\alpha = 16$, Re $= 11 \times 10^6$

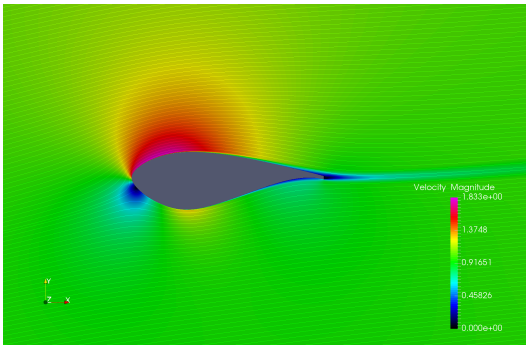


(d) DU91-W2-401, $\alpha = 24$, Re $= 11 \times 10^6$

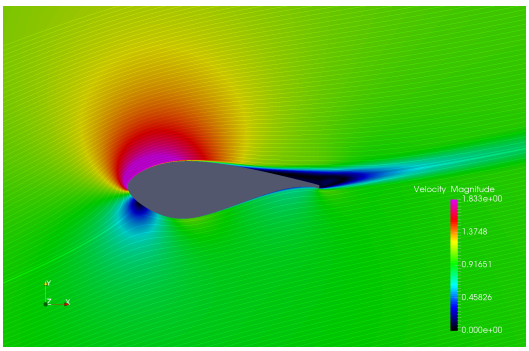
FIGURE 4.10: Velocity Magnitude contours with streamlines: DU91-W2-401



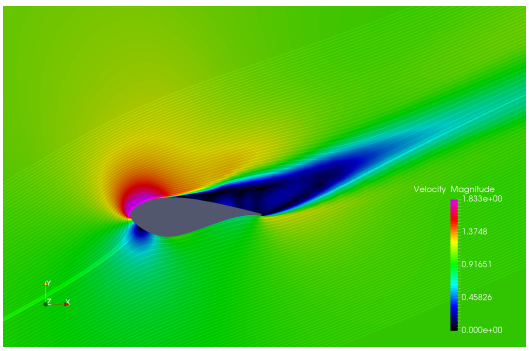
(a) DU97-W-300, $\alpha = 0$, Re $= 17 \times 10^6$



(b) DU97-W-300, $\alpha = 8$, Re $= 17 \times 10^6$



(c) DU97-W-300, $\alpha = 16$, Re $= 17 \times 10^6$



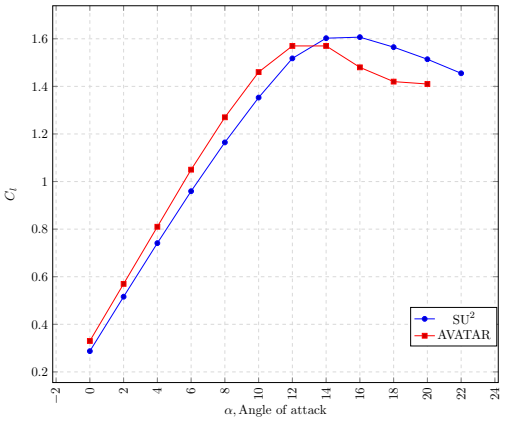
(d) DU97-W-300, $\alpha = 24$, Re $= 17 \times 10^6$

FIGURE 4.11: Velocity Magnitude contours with streamlines: DU97-W-300

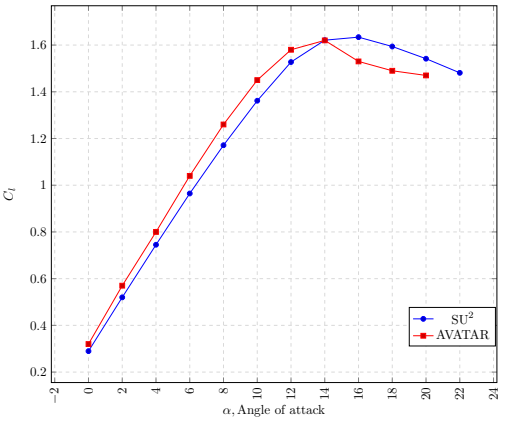
4.2 Comparison with results obtained by AVATAR partners

The Figures 4.12 and 4.13 show the lift and drag curves for the DU-00-W-212 and the DU-00-W2-250 airfoils. It must be noted that the AVATAR results taken here are the results averaged through various tools by the different AVATAR partners. Hence the behaviour is not very smooth. The following observations can be made:

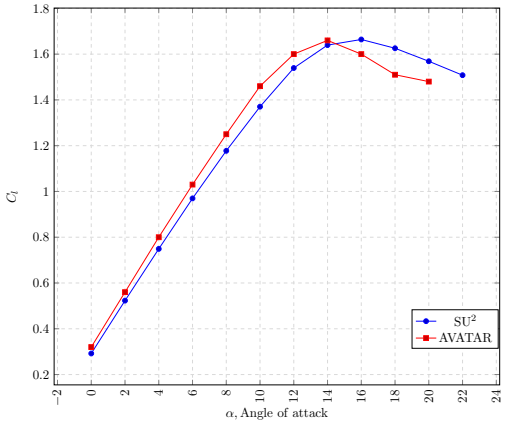
1. The behaviour of the lift and drag curves from the AVATAR results are very similar and have only a offset compared to the SU^2 results.
2. As the Reynolds number increases the behaviour of both the methods become closer to one another. This can be observed for both the DU-00-W-212 and the DU-00-W2-250 airfoils and can be observed in Figures 4.12(c) and 4.12(f).
3. As before, the AVATAR (averaged) results predict stall earlier than SU^2 does. The angle of attack at which stall is predicted though is larger than what RFOIL or XFOIL does. But this may be due to the fact that the AVATAR results being an average of the different AVATAR partners, may also contain XFOIL and RFOIL results which might have polluted the results from the pure CFD solvers.
4. The AVATAR results show a clear increase in the slope of the drag curve which is higher than from SU^2.
5. Initially SU^2 over-predicts C_d but after stall the slope of the drag curve is lower than that of AVATAR.



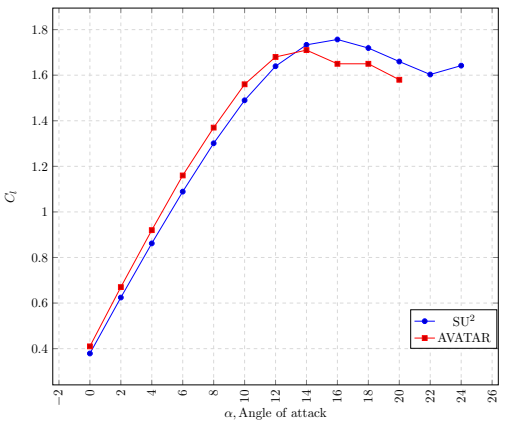
(a) DU-00-W-212 at Re = 13×10^6



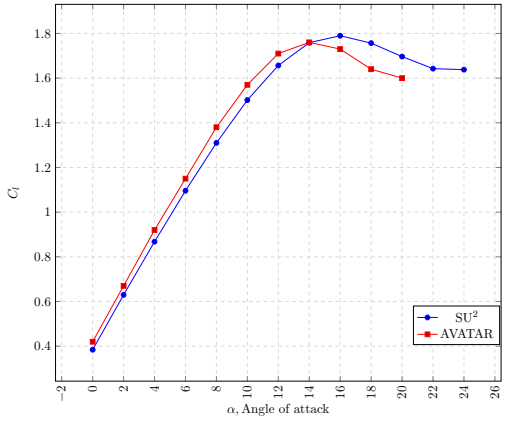
(b) DU-00-W-212 at Re = 16×10^6



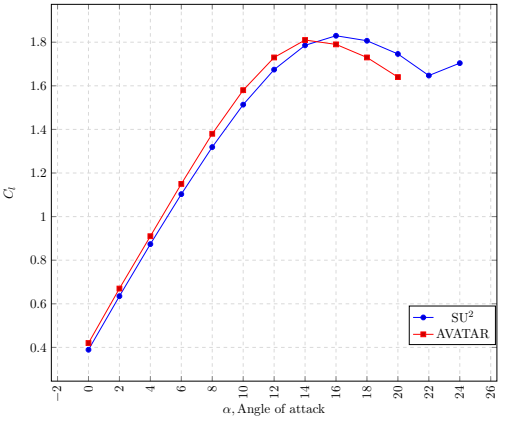
(c) DU-00-W-212 at Re = 20×10^6



(d) DU-00-W2-250 at Re = 13×10^6

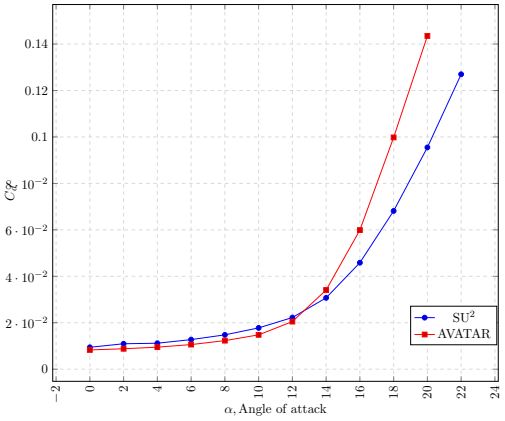


(e) DU-00-W2-250 at Re = 16×10^6

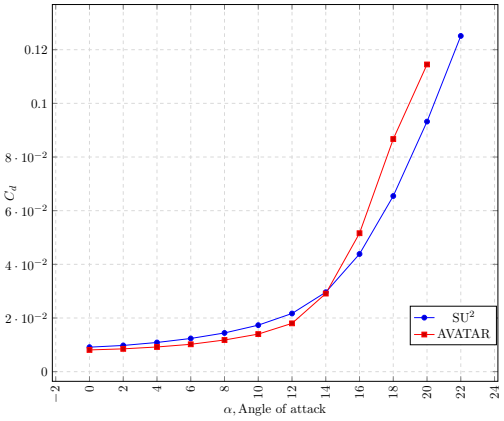


(f) DU-00-W2-250 at Re = 20×10^6

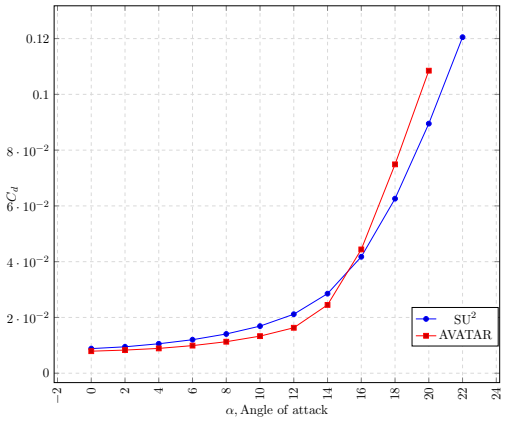
FIGURE 4.12: Comparison of C_l characteristics of airfoils with AVATAR: Part 1



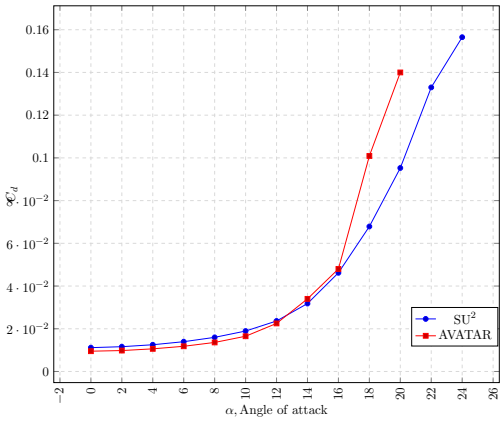
(a) DU-00-W-212 at Re = 13×10^6



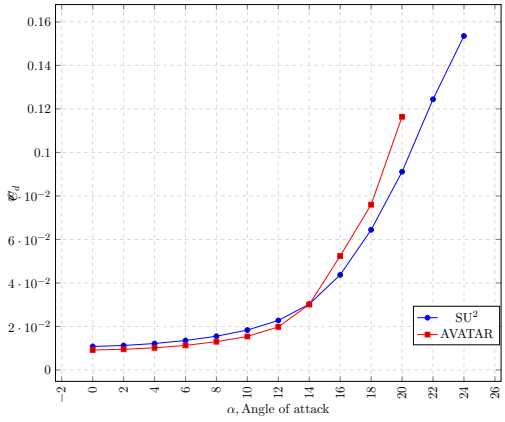
(b) DU-00-W-212 at Re = 16×10^6



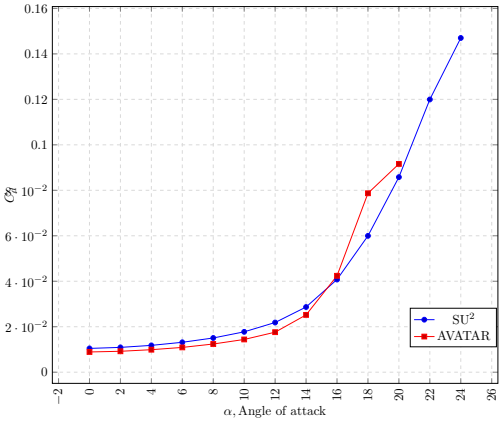
(c) DU-00-W-212 at Re = 20×10^6



(d) DU-00-W2-250 at Re = 13×10^6



(e) DU-00-W2-250 at Re = 16×10^6



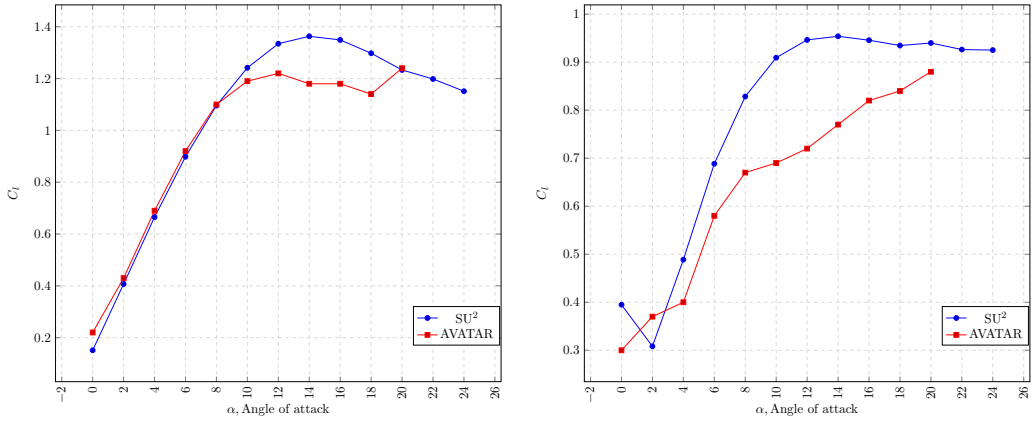
(f) DU-00-W2-250 at Re = 20×10^6

FIGURE 4.13: Comparison of C_d characteristics of airfoils with AVATAR: Part 1

The Figures 4.14 and 4.15 show the lift and drag curves for the other airfoils. The following observations can be made:

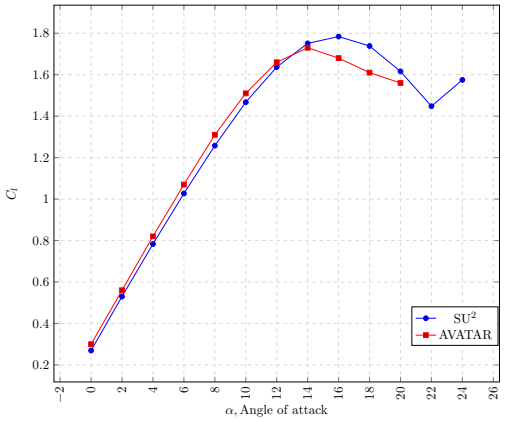
1. **For DU-00-W2-350**: It can be observed that in the linear region there is a very good agreement with the AVATAR results. After stall, the AVATAR results flatten out, which is unphysical. This is due to the averaging of the results from the different tools. The drag curve also shows a similar behaviour.

2. **For DU-91-W2-401**: This airfoil has maximum thickness which is 40.1% of the chord. This makes it a very difficult airfoil to solve using the viscous-inviscid interaction method. Hence, as observed in the graph, the results from the AVATAR are very poor with no proper observable trend. This may also be attributed to the fact that the AVATAR results also include the RFOIL results which have been shown previously to be unreliable for these type of airfoils.



(a) DU-00-W2-350 at Re $= 14 \times 10^6$

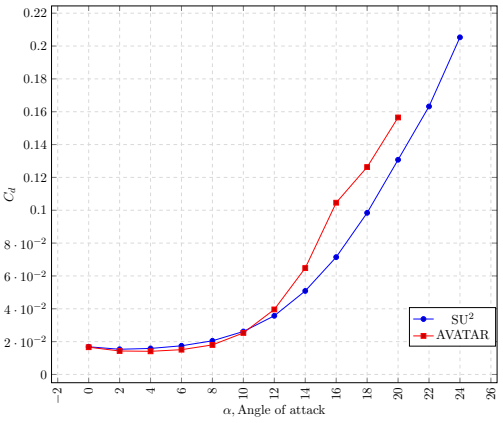
(b) DU-91-W2-401 at Re $= 11 \times 10^6$



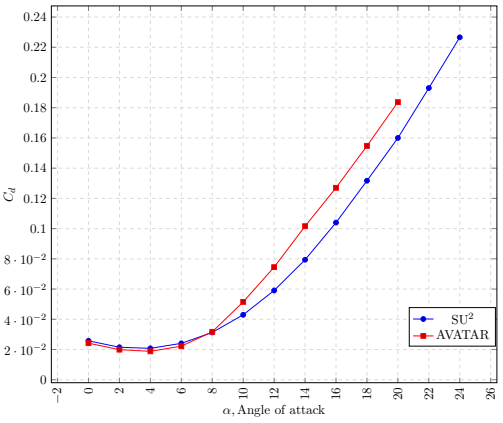
(c) DU-97-W-300 at Re $= 17 \times 10^6$

FIGURE 4.14: Comparison of C_l characteristics of airfoils with AVATAR: Part 2

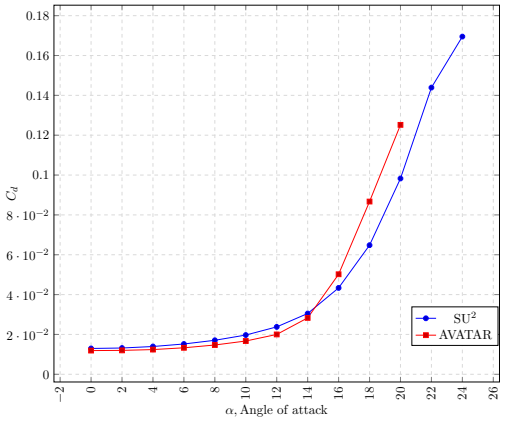
3. **For DU-97-W-300**: It can be seen that till stall, the results of both SU^2 and AVATAR match very well. Even after stall the behaviour is similar.



(a) DU-00-W2-350 at Re $= 14 \times 10^6$



(b) DU-91-W2-401 at Re $= 11 \times 10^6$



(c) DU-97-W-300 at Re $= 17 \times 10^6$

FIGURE 4.15: Comparison of C_d characteristics of airfoils with AVATAR: Part 2

Chapter 5

Conclusion and future work

This chapter provides a summary of the results and observations and notes some pitfalls that need to be avoided in CFD simulations of an airfoil.

5.1 Summary and conclusions

It can be concluded that SU^2 can be used for solving flow around airfoils. Being a CFD solver that solves the Navier Stokes with a turbulence model, it captures flow physics that may not have been possible through XFOIL and RFOIL. The following conclusions can be drawn from the observations of the previous chapters.

1. It was verified that SU^2 can be successfully used as a reliable CFD solver. The results were validated for various cases:
   (a) NACA 0012 airfoil: Validated with RFOIL and XFOIL at high Reynolds number of 13 million, Validated with experimental results at a Reynolds number of 3 million.
   (b) DU-00-W-212 airfoil: Validated with experimental results for a high Reynolds number of 15 million.
2. The mesh independence showed that beyond a certain refinement level of mesh, the solution did not change and this refinement level would be accurate enough while being computationally efficient.
3. Comparison of SU^2 results for the AVATAR airfoils with RFOIL and XFOIL showed that in the linear region the results of all the three tools were similar. In the region of stall and post-stall, there were different behaviours for different airfoils as explained in Section 4.1.

4. Comparison of results with AVATAR averaged showed that as before, the results were similar in the linear region and near stall there were differences due to the difference in the tools used for obtaining the results. This has been explained in Section 4.2.

Some recommendations based on the experience of this project are:

1. The Turbulence models and the methods used for CFD simulation can significantly affect the convergence rate and the accuracy of the solution. In some cases, it might be possible to reason the method which might be suitable for the flow problem but sometimes the effect of different combinations may result in divergence of solution which might be unexpected.

2. Meshing forms one of the most important aspects of CFD and an good mesh helps not only with giving an accurate solution but also can reduce the time required for computation significantly. This becomes very important when many runs for different cases are to be performed or when the computational resources available are limited.

3. When the geometry of the flow is simple, it is always advisable to use a structured mesh. One must also keep in mind that different mesh types may affect the solutions in different ways and hence it is possible to obtain completely varied results for two different types of meshes for the same geometry.

4. Mesh quality is an important aspect that has to be taken into consideration. Particularly the aspect ratio and the skewness of the cells must always be monitored. Sharp trailing edge airfoils, when meshed with a normal extruding O grid may produce a high skewness, which can be reduced by constructing a wake line from the trailing edge and converting the mesh into a C grid.

5. Boundary conditions are always important in a CFD solver. They should make physical sense and when a far-field condition is used, it must be made sure that the artificially induced boundary should be placed sufficiently far from the geometry.

6. For a polar simulation, for a high angle of attack, when the initial solution is the solution of the previous angle of attack, it has been observed that the accuracy is higher and also the convergence is faster than if the initial solution was the same as for the zero angle of attack.

7. The non-dimensionalised wall distance, y^+ must always be monitored and made sure that it is within the required limits so that the models used give accurate results.

5.2 Future work

The following work can be carried out in the future:

1. Conversion of the SU^2 code from a compressible solver using artificial incompressibility to a complete incompressible solver.
2. Unsteady simulations may be performed to get a more accurate result of the flow around the airfoil at higher angles of attack, particularly beyond stall when there is full separation of flow.
3. Inclusion of a transition model into the SU^2 code so that the results are comparable to that of experimental results as fully turbulent flows are generally not observed in airfoils.

Appendix A

Codes and supplements

```
1
3
5 !!!!Reads a coordinate file and writes segment files that are readable in
    Pointwise

7       program test
  real,allocatable :: a(:,:)
9   real :: pair(2)
  integer :: unit,n,newn,start,endf,endf2
11  character (len = 150) :: filen

13  unit = 11

15  read(*,*) filen
  open(unit,file=filen(1:len_trim(filen))//".pro") !Enter the name of the
    file with the x and y coordinates
17  print*, filen
  n = 0
19  do
    read(unit,*,iostat=io) pair
21    if (io/=0) exit
    n = n + 1
23  end do
  print*, n   !Prints the total number of lines in the file
25  rewind(unit)

27  allocate(a(n,2))

29  do i=1,n
    read(unit,*) a(i,:)
```

```
31      if ((a(i,2)<=4.5e-5) .and. (a(i,1)<=4.5e-5)) then
            endf = i
33          print*, endf
        else
35
        end if
37  end do

39  close(unit)


41
    open(12,file=filen(1:len_trim(filen))//".dat")  ! Enter the name of the
      file that you want to the segments to be written to.
43  write(12,*) endf
    do i=1,n
45    if (i==1) then
            write(12,*) a(i,:),0
47  else
     if(((a(i+1,2)<=4.5e-5).and.(a(i+1,1)<=4.5e-5)).and.(i/=n))then
49      write(12,*) a(i,:),0
        write(12,*) a(i+1,:),0
51      write(12,*) n-endf+1
     else
53      write(12,*) a(i,:),0
          end if
55   end if
    end do
57  if (a(1,2)/=0.0) then
        write(12,*) 2
59      write(12,*) a(n,:),0
        write(12,*) a(1,:),0
61  end if


63      end
```

Script used to run polar simulation in SU2:

```
#!/bin/bash
#Runs SU2 for polar cases taking the previous angle of attack as the input


PI="3.141592653"
declare -i initAoa
declare -i maxAoa
proc="24"
read -t3 -e -i "$proc" -p "Enter number of processes for SU2 to be run: "
    input
proc="${input:-$proc}"
echo "Will run on $proc processes!"

read -e -p "Enter name of the configuration file with .cfg extension: " -i
    "naca0012.cfg" cfg

read -e -p "Enter name of the mesh file with .su2 extension: " -i "
    mesh_NACA0012_turb_449x129.su2" mesh

echo "Will replace 'MESH_FILENAME=*' with 'MESH_FILENAME=$mesh'"
meshrep="MESH_FILENAME"

#Write the mesh file name
/home/nas11/nayak/Documents/004scripts/f_replace_example $cfg $mesh
    $meshrep


#Angles of attack
read -e -p "Enter initial angle of attack: " -i "0" initAoa

read -e -p "Enter maximum angle of attack: " -i "15" maxAoa

free_vel="FREESTREAM_VELOCITY"

#Reynolds Number
read -e -p "Enter Reynolds number desired: " -i "13.0e6" Re

#Viscosity
free_visc="FREESTREAM_VISCOSITY"
Re="($(sed 's/[eE]+\{0,1\}/*10^/g' <<<"$Re"))"
visc=$(bc -l <<< "1.0/$Re")

echo "Will replace 'FREESTREAM_VISCOSITY=*' with 'FREESTREAM_VISCOSITY=
    $visc'"
/home/nas11/nayak/Documents/004scripts/f_replace_example $cfg "$visc"
    $free_visc
```

```
41
#X and Y velocities
43 x_vel=$(bc -l <<< "c($initAoa*$PI/180.0)")
y_vel=$(bc -l <<< "s($initAoa*$PI/180.0)")
45
/home/nas11/nayak/Documents/004scripts/f_replace_example $cfg "($x_vel,
    $y_vel,0.0)" $free_vel
47
#Run the first angle of attack = ZERO
49 mpirun -n $proc SU2_CFD $cfg

51 #Setting RESTART_FLOW to YES
restart_flow="RESTART_SOL"
53
echo "Will replace '$restart_flow=NO' with '$restart_flow=YES'"
55 /home/nas11/nayak/Documents/004scripts/f_replace_example $cfg "YES"
    $restart_flow


57
#Enter the next angle of attack to be computed
59 while (( initAoa < maxAoa )); do
    Aoan="2"
61  read -t3 -e -i "$Aoan" -p "Enter the increment or decrease in angle of
    attack required from $initAoa ,(int value) : " input
    Aoan="${input:-$Aoan}"
63  echo "$Aoan"
    Aoan=$(bc -l <<< "$initAoa+$Aoan")
65
    cd ..
67
    mkdir $Aoan
69
    cd ./$Aoan
71
    cp ../$initAoa/restart_flow.dat .
73  cp ../$initAoa/$mesh .
    #rm ../$initAoa/$mesh
75  cp ../$initAoa/$cfg .

77  #Set angle of attack and velocities
    x_vel_new=$(bc -l <<< "c($Aoan*$PI/180.0)")
79  y_vel_new=$(bc -l <<< "s($Aoan*$PI/180.0)")

81  echo "Will replace '$free_vel=*' with '$free_vel= ($x_vel_new,
    $y_vel_new,0.0) "
    /home/nas11/nayak/Documents/004scripts/f_replace_example $cfg "(
    $x_vel_new,$y_vel_new,0.0)" $free_vel
83
```

```
    #Run the ith angle of attack
85      mpirun -n $proc SU2_CFD $cfg

87      initAoa=$Aoan

89  done
```

Bibliography

[1] SU^2 wiki. URL https://github.com/su2code/SU2/wiki. Accessed: 14:00, 28th April, 2016.

[2] Thomas D. Economon et al. SU^2: An open-source suite for multiphysics simulation and design. *AIAA Journal*, Vol 54, Issue 3:828–846, March 2016. URL http://arc.aiaa.org/doi/abs/10.2514/1.J053813.

[3] Mark Drela. *Low Reynolds Number Aerodynamics: Proceedings of the Conference Notre Dame, Indiana, USA, 5–7 June 1989*, chapter XFOIL: An Analysis and Design System for Low Reynolds Number Airfoils, pages 1–12. Springer Berlin Heidelberg, Berlin, Heidelberg, 1989. ISBN 978-3-642-84010-4. doi: 10.1007/978-3-642-84010-4_1. URL http://dx.doi.org/10.1007/978-3-642-84010-4_1.

[4] Giridhar Ramanujam, Hüseyin Özdemir, and Harry W Hoeijmakers. Improving airfoil drag prediction. *AIAA,34th Wind Energy Symposium, San Diego, CA, USA*, 2016. URL http://arc.aiaa.org/doi/abs/10.2514/6.2016-0748.

[5] F.R Menter. Zonal two equation $k-\omega$ turbulence models for aerodynamic flows. *AIAA paper*, July 1993. URL http://dx.doi.org/10.2514/6.1993-2906.

[6] P.L. Roe. Approximate reimann solvers, parameter vectors, and difference schemes. *Journal of Computational Physics*, 43:357–372, October 1981. URL http://www.sciencedirect.com/science/article/pii/0021999181901285.

[7] Alexandre Joel Chorin. A numerical method for solving incompressible viscous flow problems. *Journal of Computational Physics*, 135(2):118 – 125, 1997. ISSN 0021-9991. doi: http://dx.doi.org/10.1006/jcph.1997.5716. URL http://www.sciencedirect.com/science/article/pii/S0021999197957168.

[8] N Gregory and C.L.O'Reilly. Low-speed aerodynamic characteristics of naca aerofoil section, including the effects of upper-surface roughness simulating

hoar frost. *Aeronautical Research Council, Ministry of Defence, UK*, Reports and Memoranda No. 3726, January 1970. URL http://naca.central.cranfield.ac.uk/reports/arc/rm/3726.pdf.

[9] Hui Hu and Zifeng Yang. An experimental study of the laminar flow separation on a low-reynolds-number airfoil. *Journal of Fluids Engineering*, 130(5):051101, 2008.