

# **FOSSEE Case Study Project**

**CFD with OpenFOAM**

**Numerical study of flow of electrically conducting fluid over  
an airfoil**

**Developed on OpenFOAM - v1812**

**Author:**

**Rahul Radhakrishnan**

**Final year Mechanical Engineering student**

**Vellore Institute of Technology, Vellore**

## About the software used

Ubuntu Bash was activated on Windows 10. OpenFOAM – v1812 was extracted and compiled following the procedure given in <https://www.openfoam.com/download/install-windows-10.php>

Paraview 5.2.0 for windows was installed to visualize OpenFOAM results. Gmsh 4.3.0 and ANSYS ICEM 18.1 were used for generating the mesh.

Microsoft Excel was used to generate the plots using the data obtained from the simulations.

# Content

1. Custom Solver: Introduction to *mhdTurbFoam*
  - 1.1 Extension to *mhdFoam*
  - 1.2 Setting up the solver
  - 1.3 Modifying the turbulence models
2. Validation
  - 2.1 Parameters
  - 2.2 Geometry and Meshing
  - 2.3 Boundary conditions and setting up the case files
  - 2.4 Getting the results
  - 2.5 Results
  - 2.6 Conclusion
3. Case Study
  - 3.1 Parameters
  - 3.2 Geometry and Meshing
  - 3.3 Boundary conditions and setting up the case files
  - 3.4 Results
  - 3.5 Conclusion
4. References

# Introduction to *mhdTurbFoam*

## 1.1 Extension to *mhdFoam*

*mhdFoam* is an OpenFOAM incompressible, laminar magnetohydrodynamic equation solver. It solves the coupled Maxwell-Navier-Stokes equations for an incompressible electrically conducting fluid. In this section, we will discuss to understand what MHD flows are and how to model turbulence for such flows.

Fluid dynamics governing equations:

$$(1) \quad \nabla \cdot \mathbf{U} = 0$$

$$(2) \quad \frac{D\mathbf{U}}{Dt} = -\nabla \left( \frac{P}{\rho} \right) + \nu \nabla^2 \mathbf{U} + \frac{(\mathbf{J} \times \mathbf{B})}{\rho}$$

Where  $\mathbf{U}$  is the fluid velocity,  $\mathbf{J}$  is the current density and  $\mathbf{B}$  is the magnetic field flux.

Maxwell's governing MHD equations:

$$(3) \quad \frac{\partial \mathbf{B}}{\partial t} = \nabla \times (\mathbf{U} \times \mathbf{B}) + \lambda \nabla^2 \mathbf{B}$$

$$(4) \quad \nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t}$$

$$(5) \quad \nabla \times \mathbf{B} = \mu \mathbf{J}$$

$$(6) \quad \nabla \cdot \mathbf{J} = 0$$

$$(7) \quad \nabla \cdot \mathbf{B} = 0$$

Here,  $\mathbf{E}$  is the electric field strength and  $\lambda$  is fluid property equal to  $1/\sigma\mu$ , where  $\sigma$  is the electrical conductivity of the fluid and  $\mu$  is magnetic permeability. It should be noted that the displacement current and the charge density in equations (5) and (6) respectively are omitted because they play no major role in such flows ( $R_m \ll 1$ ).

The volume force can be further split into three components:

$$(8) \quad \mathbf{J} \times \mathbf{B} = \frac{\partial}{\partial s} \left[ \frac{B^2}{2\mu} \right] \mathbf{t} - \frac{B^2}{\mu R} \mathbf{n} - \nabla \left( \frac{B^2}{2\mu} \right)$$

The first two terms represent the Maxwell's forces acting along tangential and normal direction along the field lines and the third term is called the magnetic pressure. If a fluid is

moving across a perpendicular field, then the tangential component tries to transfer momentum away from the centre and the normal component tries to stop the fluid from moving further. As we can see, the significance of these forces depends on the electrical conductivity of the fluid.

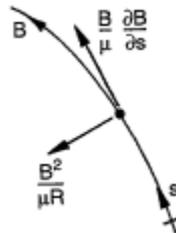


Fig. 1 Forces acting normal and tangential to the field lines.

For a turbulent flow, the Navier-stokes equation can be modified as,

$$(9) \quad \frac{DU}{Dt} = -\nabla \left( \frac{P}{\rho} \right) + \nu_{eff} \nabla^2 \mathbf{U} + \frac{(\mathbf{J} \times \mathbf{B})}{\rho}$$

Where  $\nu_{eff}$  is the effective viscosity which includes the effect of turbulence. Various turbulence models are made to predict the value of turbulent viscosity and hence, helps in closure of the entire problem.

It appears so that if at all any disturbance was introduced in MHD flows, the effect of magnetic field would be to try to reduce it. In fact, it is a well-known concept that such a process will lead to relaminarization and effective control of the flow. There were many studies performed to introduce the effect of turbulence in incompressible MHD flows. Most of them were done to modify Spalart-Allmaras and k-Epsilon RANS turbulence models. Smolentsev [2] has discussed all the modifications required to apply the k-Epsilon models to such flows.

$$\frac{\partial K}{\partial t} + \langle v_j \rangle \frac{\partial K}{\partial x_j} = \underbrace{\nu_t \left( \frac{\partial \langle v_i \rangle}{\partial x_j} \right)^2}_{\text{Production}} + \underbrace{\frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\nu_t}{\sigma_K} \right) \frac{\partial K}{\partial x_j} \right]}_{\text{Diffusion}} - \underbrace{\left( \varepsilon - \varepsilon_{em}^K \right)}_{\text{Dissipation}}.$$

$$\frac{\partial \varepsilon}{\partial t} + \langle v_j \rangle \frac{\partial \varepsilon}{\partial x_j} = C_1 \frac{\varepsilon}{K} \nu_t \left( \frac{\partial \langle v_i \rangle}{\partial x_j} \right)^2 + \frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\nu_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_j} \right] - C_2 \frac{\varepsilon}{K} \varepsilon - \varepsilon_{em}^\varepsilon.$$

Fig 2. Modified k-Epsilon equation for MHD flows

The modifications were to introduce a magnetic term which sequentially helps reducing the turbulent viscosity. Here, in the above equation, the electromagnetic term (em) has been added to the original k-Epsilon model equations. Once, the above equations are solved,

the value of  $k$  and  $\epsilon$  is used to solve Boussinesq eddy viscosity assumption. The details of the calculation of the additional term can be obtained from Smolentsev's paper. However, what is important to know is that, in this case the additional term depends on the characteristic length ( $L$ ) of the domain.

Dietiker [3] modified the one-equation Spalart-Allmaras model. The original model states that,

$$\begin{aligned} \frac{D\bar{v}}{Dt} &= C_{b1}(1 - f_{t2})\bar{S}\bar{v} + \frac{1}{\sigma} \{ \nabla[(v + \bar{v})\nabla\bar{v}] + C_{b2}(\nabla\bar{v})^2 \} \\ &+ \frac{C_{b1}}{\kappa^2} f_{t2} \left( \frac{\bar{v}}{d} \right)^2 - C_{w1} f_w \left( \frac{\bar{v}}{d} \right)^2 + f_{t1} (\Delta q)^2 \\ f_{v1} &= \chi^3 / (\chi^3 + C_{v1}^3) \end{aligned}$$

Fig 3. Spalart-Allmaras one equation model

Though, there are other equations required to close the first equation we will restrict ourselves to these two equations above. In the model,  $C_{v1}$  is taken as a constant value equal to 7.1. The modified equation for  $C_{v1}$  is:

$$(10) \quad C_{v1} = 7.1 \min \left\{ 2.6 + 1.6 \tanh \left[ 4430 \frac{Ha}{Re} - 19.345 \right] + 22 \frac{Ha}{Re}; 4.225 \right\}$$

Where,  $Ha$  is Hartmann number  $\left( \sqrt{\frac{\sigma L^2 B^2}{\rho \nu}} \right)$  and  $Re$  is the Reynold's number  $\left( \frac{UL}{\nu} \right)$ .

Here, no additional magnetic term is added to the partial differential equation governing the viscosity ( $\nu$ ). Rather, the effect of the field is accounted for within the closure coefficient which becomes a function of the magnetic field. Dietiker's motivation behind this approach lies in that the Spalart-Allmaras model performs the best in the prediction of the relaminarization of the MHD Hartmann flow (which is available as a tutorial case for *mhdFoam*). For a certain value of  $Ha/Re$  value, the flow completely laminarizes and further modification is no longer necessary. That is the reason why we have a threshold for  $C_{v1}$  in the above equation, after which it remains constant. Experiments performed by Brouillete [4] predict this value to be about 1/225.

So, in conclusion, if one needs to modify *mhdFoam* to incorporate turbulence, then we need bring in the turbulent Navier-Stokes equation into the solver and make additional changes to the turbulence model that one desires to use as per discussed above. In this particular case study, we will be focusing on the modified Spalart-Allmaras model.

## 1.2 Setting up the solver

In this section, we will be seeing how to set up the *mhdTurbFoam* solver in OpenFOAM.

1. Since *mhdFoam* (can be obtained in the electromagnetic solver directory) is the closest solver which resembles our new solver, we make a copy of its solver files,

rename it and place it in the same directory. Rename all the instances where *mhdFoam* is used.

2. Now it's time to make the modifications. *createFields.H* defines all the variables used in the solver. Since we are going to introduce the turbulence term in the solver, we will add the following code to the file at the end:

```
singlePhaseTransportModel laminarTransport(U, phi);
```

```
autoPtr<incompressible::turbulenceModel> turbulence  
(  
    incompressible::turbulenceModel::New(U, phi, laminarTransport)  
);
```

3. Include the files *singlePhaseTransportModel.H* and *turbulentTransportModel.H* in the *mhdTurbFoam.C* file. Make following modifications to the fvVectorMatrix UEqn:

```
fvVectorMatrix UEqn  
(  
    fvm::ddt(U)  
    + fvm::div(phi, U)  
    + turbulence->divDevReff(U)  
    - fvc::div(phiB, 2.0*DBU*B)  
    + fvc::grad(DBU*magSqr(B))  
);
```

**Note:** The only modification done here is to add the turbulence term. This expression will point towards the effective viscosity term in eqn. (9), which in turn is obtained from turbulence modelling.

4. In the Make directory, change the name of the solver in *files*. Add the missing libraries present below in the *options* file. These are essentials for turbulence modelling.

```
EXE_INC = \  
-I$(LIB_SRC)/TurbulenceModels/turbulenceModels/InInclude \  
-I$(LIB_SRC)/TurbulenceModels/incompressible/InInclude \  
-I$(LIB_SRC)/transportModels \  
-I$(LIB_SRC)/transportModels/incompressible/singlePhaseTransportModel \  
-I$(LIB_SRC)/finiteVolume/InInclude \  
-I$(LIB_SRC)/meshTools/InInclude \  
-I$(LIB_SRC)/sampling/InInclude
```

```
EXE_LIBS = \  
-lturbulenceModels \  
-lincompressibleTurbulenceModels \  
-lincompressibleTransportModels \  
-lfiniteVolume
```

```
-lmeshTools \  
-lfvOptions \  
-lsampling
```

5. Run `./Allwmake` to compile the solver.

### 1.3 Modifying the turbulence models

As we know from the first section that modification is needed in the turbulence model in order to simulate MHD flows. It is also a good time to remember that the modified K-Epsilon model relies on the characteristic length (L) of the domain, which may be difficult if one is trying to simulate multiple cases, as I have done in this project. However, if we observe the modification in the Spalart-Allmaras model, the closure coefficient depends on the ratio of  $Ha/Re$ . This eliminates the dependency of the model on L, which makes it easier to implement in OpenFOAM than the modified k-Epsilon model.

Since the only modification that is needed in the model is change in closure coefficient  $C_{v1}$  for different cases which I will simulate, there is no need to change the OpenFOAM code for the model. While running a case, the  $C_{v1}$  value can be changed in the `turbulenceProperties` file in the case directory as follows:

```
simulationType RAS;  
  
RAS  
{  
  RASModel    SpalartAllmaras;  
  SpalartAllmarasCoeffs  
  {  
    Cv1 7.185;  
  }  
  turbulence on;  
  printCoeffs on;  
}
```

This will automatically change the closure coefficient value while running the simulation.

# Validation

Now before we jump into the project using this turbulence model, it is important to understand the reliability of the model in OpenFOAM. We need to know if OpenFOAM can give accurate results if the modifications in the solver and turbulence model is incorporated.

For this case study's validation, we will be replicating the study performed by Dietiker on turbulent Hartmann flow.

## 2.1 Parameters

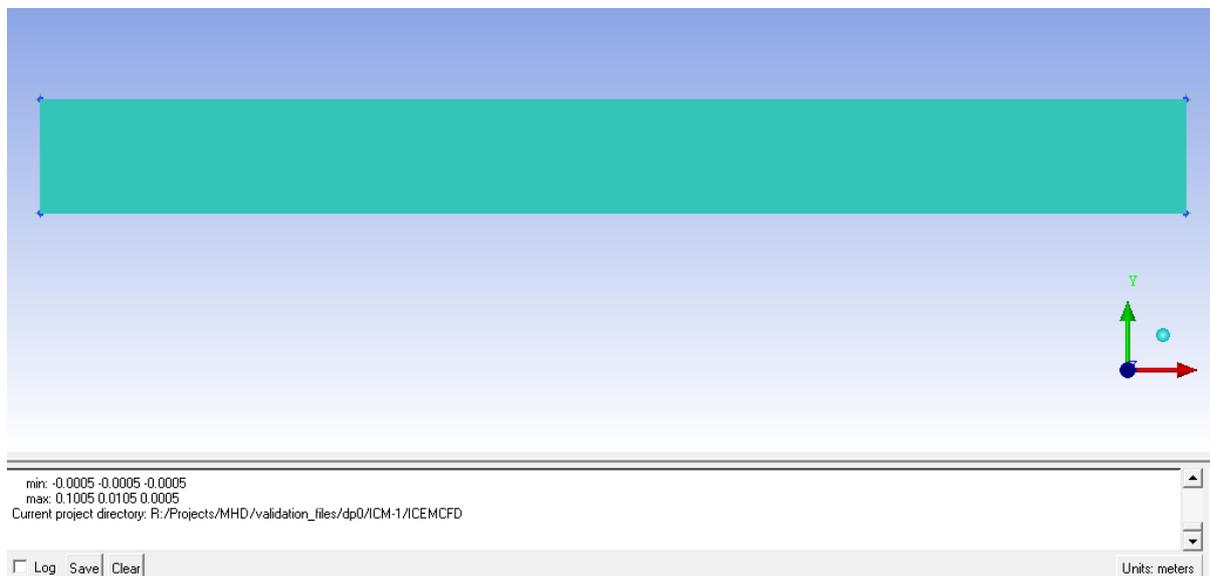
Hartmann flow refers to fully developed flow between two parallel plates subjected to a magnetic field perpendicular to the direction of the flow. The ratio between the length and the height of the domain is 10. The Reynolds number used for the problem is based on half-height between the two plates. The flow conditions are mentioned in the table below.

Property	Value
Density	1.225 Kg/m <sup>3</sup>
Electrical conductivity	800 S/m
Kinematic Viscosity	1.47E-05 m <sup>2</sup> /s
Half-height between the plates	0.005 m

Table 1. Parameters involved in the MHD Hartmann flow

## 2.2 Geometry and Meshing

A 0.1 m × 0.01 m 2-D geometry was made and meshed using ANSYS ICEM package. The domain was divided into 10 nodes in the x-direction along the length and 150 nodes in the y-direction as mentioned in [3]. The grid was clustered near the wall to get an accurate prediction of the Hartmann layer.



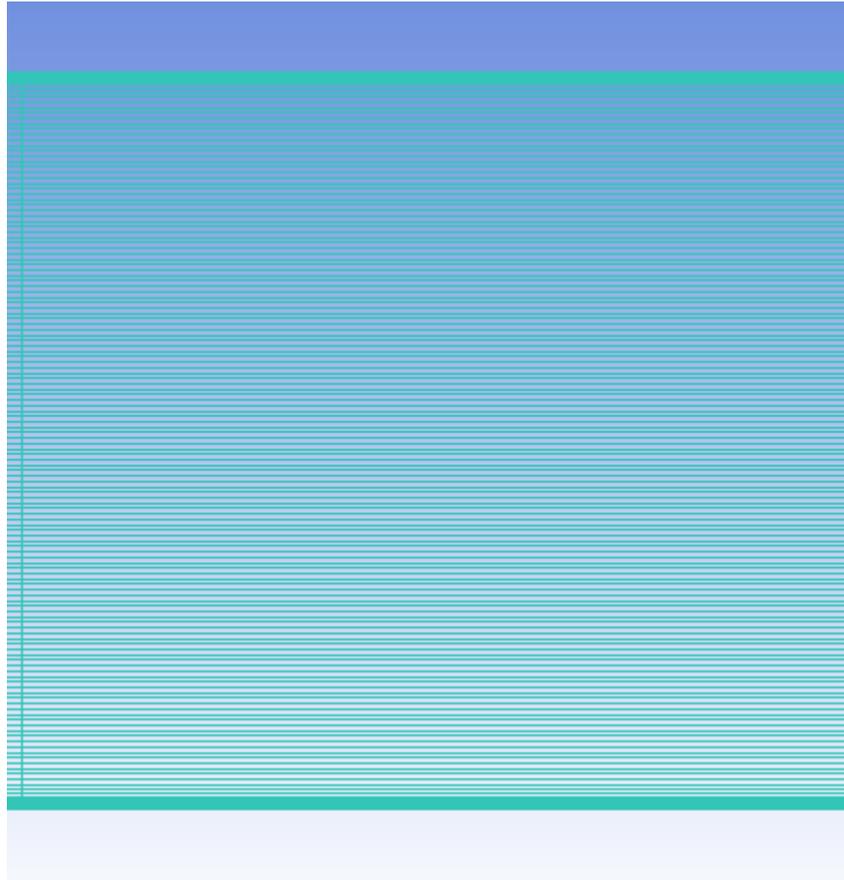


Fig. 4 Geometry and Meshing for MHD Hartmann flow

## 2.2 Boundary conditions and setting up the case files

The simulation is to be performed over Reynold's number ranging from 5000 to 50000. The left boundary is treated as velocity inlet, right boundary as pressure outlet, the lower and upper boundary as wall. Table 2 lists all the conditions which have been used.

	<b>Inlet</b>	<b>Outlet</b>	<b>upperWall</b>	<b>lowerWall</b>
<b>U</b>	fixedValue	zeroGradient	noSlip	noSlip
<b>B</b>	zeroGradient	zeroGradient	fixedValue	fixedValue
<b>nut</b>	Calculated	Calculated	nutUSpaldingWallFunction	nutUSpaldingWallFunction
<b>nuTilda</b>	fixedValue	zeroGradient	zeroGradient	zeroGradient
<b>p</b>	zeroGradient	fixedValue	zeroGradient	zeroGradient

Table 2. *O* directory in the case directory.

Note:  $pB$  is a correction variable to compensate for error and keep the magnetic flux divergence free, hence no changes required.

As we have seen earlier, the closure coefficient keeps varying for different values of  $Ha/Re$ . A summary of values of the closure coefficient is listed in Table 3.

<b>Re</b>	<b>B (T)</b>	<b>Ha/Re</b>	<b>C<sub>v1</sub></b>
5E+03	0.1	0.666	7.204
	0.2	1.333	7.308
	0.4	2.666	7.5
	0.8	5.332	30
1E+04	0.4	1.3	7.224
	0.8	2.61	7.508
	1.2	3.92	8.134
	1.5	4.9	30
5E+04	1	0.666	7.204
	4	2.66	7.516
	6	3.99	8.566
	8	5.332	30

Table 3. Closure coefficient for the Modified Spalart-Allmaras turbulence model

Following modifications need to be made in the *system* directory:

- a) *fvSolutions*: Set solver for  $p$  as GAMG with GaussSeidel as a smoother. It was observed that GAMG solver was much more stable than PCG for a given time step value. Further, PCG required lesser time step value which would have increased the computation time. Add *nuTilda* to the list too and set the smoother for U, B and nuTilda as GaussSeidel.
- b) *fvSchemes*: The following should be added to the divergence schemes to incorporate the turbulence term and the magnetic field equation term (eqn 3):

```
div(phiB,U) Gauss limitedLinear 1;
div(phi,B) Gauss limitedLinear 1;
div(phi,nuTilda) Gauss limitedLinear 1;
div(phiB,((2*DBU)*B)) Gauss linear;
div((nuEff*dev2(T(grad(U)))) Gauss linear;
```

The limitedLinear coefficient is set to 1 to limit the scheme towards upwind scheme (first order).

## 2.3 Getting the Results

*mhdturbFoam* is a transient solver like *mhdFoam*. In order to obtain values like skin friction coefficient, it is important that we take steady state values into our calculations. For this reason, the simulations were run for a long enough time to the point where there were not too much fluctuations in the result. In this case, it was observed that 1 second was a reasonable amount of time to perform the simulation. The time step needs to be varied and kept just enough so that the courant number remains below 1 to ensure stability of the solution.

In order to obtain the skin friction coefficient, we need to follow the procedure as given below:

1. Once the simulation is complete, we need to obtain the wall shear stress acting on the upper and lower walls. This can be obtained by the following command:

```
mhdTurbFoam -postProcess -func wallShearStress
```

This generates a folder inside the case directory named *postProcessing* which contains a file named *wallShearStress.dat*. This has the minimum and maximum shear stress values for different time.

2. The *controlDict* file is to be imported in Paraview 5.2.0 to observe the pressure and velocity contours.
3. Skin friction coefficient is governed by the following equation:

$$(11) \quad C_f = \frac{\tau}{\frac{1}{2}\rho U^2}$$

A calculator should be where this equation needs to be added. It needs to be noted that in the above formula,  $\rho$  and  $U$  are the values of density and velocity of the fluid respectively at the inlet.

## 2.4 Results

Simulations were performed to obtain the contours for all three Reynolds number and for different  $Ha/Re$  ratios.



(a)  $Ha/Re = 0$

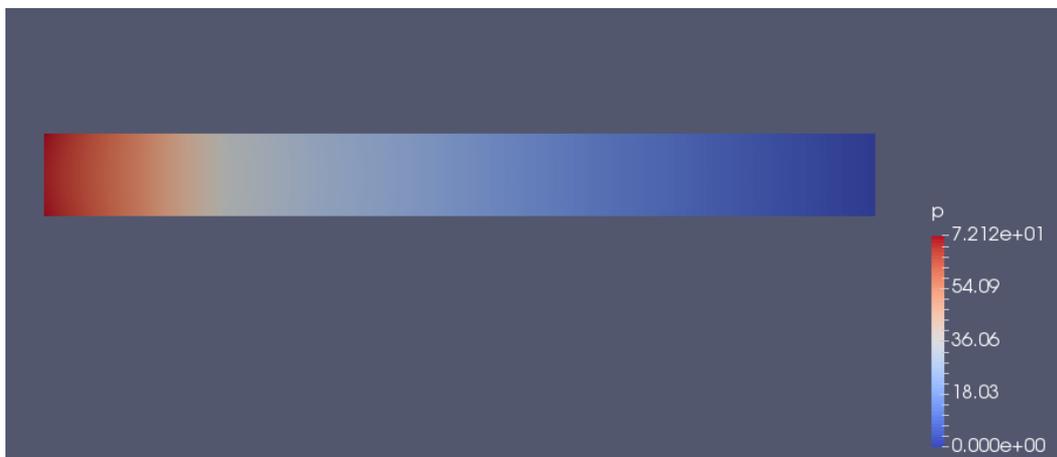


(b)  $Ha/Re = 3.26$

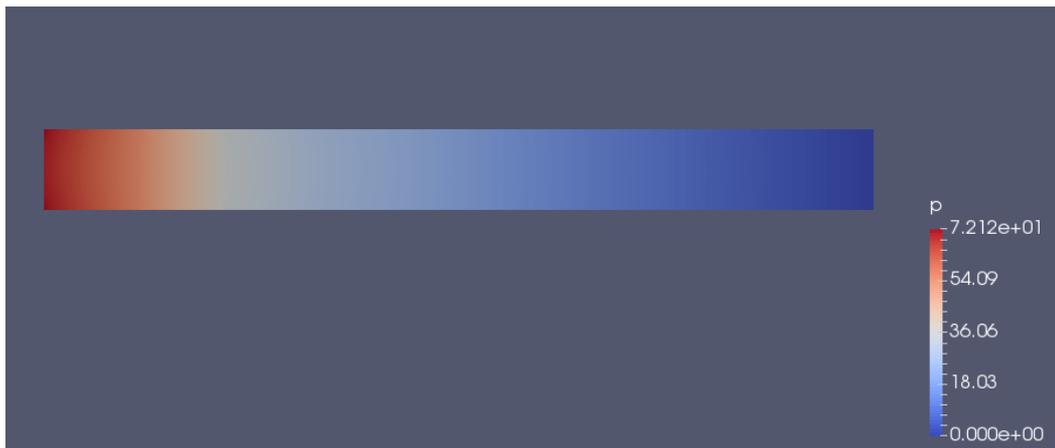


(c)  $Ha/Re = 13$

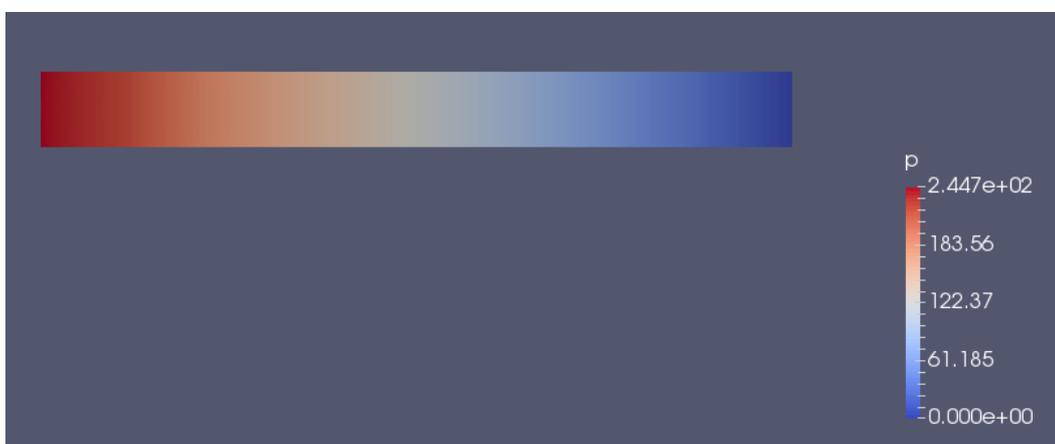
Fig. 5 Velocity contours of the turbulent Hartmann flow at  $Re = 1E+04$



(a)  $Ha/Re = 0$



(b)  $Ha/Re = 3.26$



(c)  $Ha/Re = 13$

Fig. 6 Pressure contours of the turbulent Hartmann flow at  $Re = 1E+04$ .

<b>Re</b>	<b>Ha/Re*1000</b>	<b>C<sub>f</sub>*1000</b>
<b>5E+03</b>	0.666	9.55
	1.333	9.737
	2.666	10.454
	5.332	12.928
<b>1E+04</b>	1.3	6.828
	2.61	7.782
	3.92	9.144
	4.9	10.34
<b>5E+04</b>	0.666	3.034
	2.66	5.224
	3.99	7.136
	5.332	9.149

Table 6. Summary of skin friction coefficient values at different Reynolds number\*

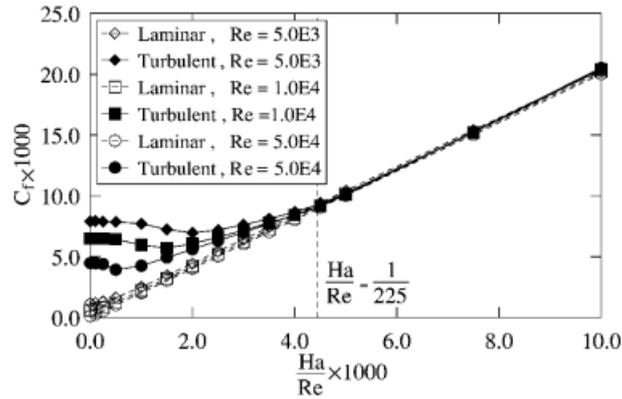


Fig. 7 Summary of skin- friction coefficient by Dietiker

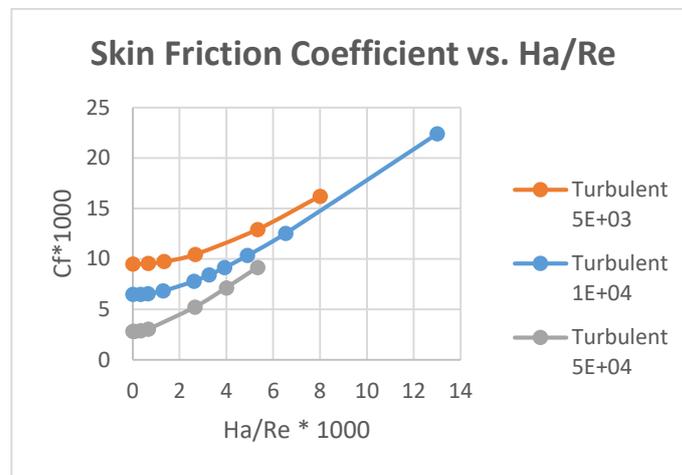


Fig. 8 Summary of skin-friction coefficient performed in OpenFOAM

\*Note: Only few results are shown in the table and in Fig. 5 and 6. Many more simulations were performed to capture the trend observed in Fig.8. Rest of the results can be found in “All countours.docx” file.

## 2.4 Conclusion

On comparing with the present simulations, we see that *mhdturbFoam* accurately describes the transition regime (turbulent to laminar –  $0 < Ha/Re < 1/225$ ) in the MHD Hartmann flow. Though the convergence of different Reynolds number lines is predicted in the relaminarization zone ( $Ha/Re > 1/225$ ), the rate of their convergence is slightly slower in *mhdturbFoam* when compared to that of in Dietiker’s. However, in our present study, we are not concerned about finding out the exact transition line. In nature, such a line does not exist but a transition regime exists where the flow is relaminarized gradually. In this aspect, *mhdturbFoam* has an advantage over it.

It is useful to note how transverse magnetic field extends the region of maximum flow velocity and also increases the pressure difference between inlet and outlet.

To conclude, *mhdturbFoam* serves its purpose very well and can be a good solver to perform turbulent MHD flows.

# Case Study

## 3.1 Abstract

Magneto and Electro aerodynamics (MAD and EAD) can open up new possibilities in the field of propulsion and power generation. Such concepts can reduce the number of moving parts in an aircraft and can help reducing the overall emissions into the atmosphere. Recently, MIT demonstrated its concept of a solid-state aircraft using EAD system. This project's aim is to study the flow of an electrically conducting fluid over NACA 0012 under the influence of magnetic field. A new solver, *mhdTurbFoam*, was made in OpenFOAM and used for this purpose. Reynolds number at which this study was performed is 1,000,000 corresponding to a velocity of 10 m/s. Enhancement in lift is observed due to flow acceleration with increasing magnetic field strength. The drag also increases with magnetic field and this method becomes infeasible at high magnetic field strength as the increment in drag becomes more predominant than increase in lift.

## 3.2 Introduction

Utilization of electro-magnetic forces to control flow is not a new concept. In the past, there have been broadly two types of studies: flow control using oscillating Lorentz forces and steady Lorentz forces. Mutschke and Gerberth [6] performed flow control study on hydrofoil (NACA 0017) using numerical simulation in the laminar regime ( $Re = 500$ ). They kept an oscillatory Lorentz force in the leading edge and steady force at the suction side, both in the streamwise direction. It was observed that the separation is easily suppressed by steady forces, turbulent vortices were eliminated and the lift was enhanced, but oscillatory forces gave better control over the lift. In another research [5], by the same authors, DNS simulations were performed over the same hydrofoil at low Reynolds number ( $3E+04$ ). It was noticed that post stall, lift increased by 90 % and drag reduced.

Sedaghat and Badri [7] performed numerical study on NACA 0015 hydrofoil at angle of attack 15 degrees to 30 degrees, region of stall by producing a steady Lorentz force in the streamwise direction. It was set at low speeds at high angles of attack. In all cases, separation was completely prohibited using the Lorentz force induced by the electromagnetic field. By increasing angles of incidence, the lift coefficient as well as the drag coefficient over the hydrofoil was increased. However, lift gain was greater than drag increase.

In this study, we will be using steady, chord parallel streamwise Lorentz force (Magnetic field is in the y-direction for all case – normal to chord) at  $Re = 10^6$  and angles of attack ranging from 0 to 25 degrees with 5 degrees of increment. The airfoil to be used is NACA 0012.

## 3.3 Parameters

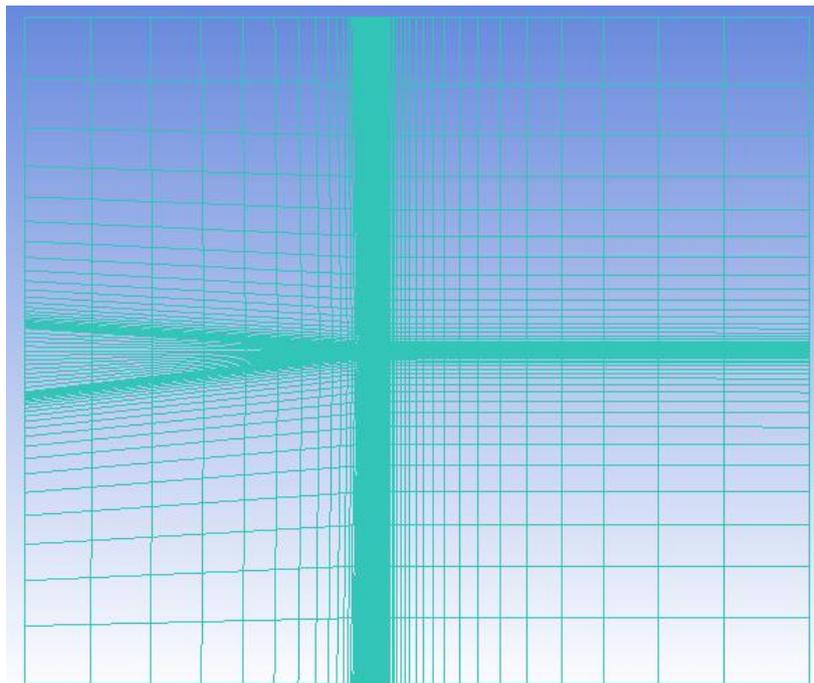
The fluid used in this study has low conductivity. The value of this conductivity lies close to that of sea saltwater. However, the values of the other parameters are chosen in a way assuming that the fluid is an ionized air and exact values of such a medium could not be found. All the values were kept in the orders of magnitude to simplify calculations. These parameters need to be added and edited in the *transportProperties* file in the case directory.

<b>Property</b>	<b>Value (in SI units)</b>
Density ( $\rho$ )	1
Magnetic permeability ( $\mu$ )	1E-06
Electrical conductivity ( $\sigma$ )	3
Kinematic viscosity ( $\nu$ )	1E-05

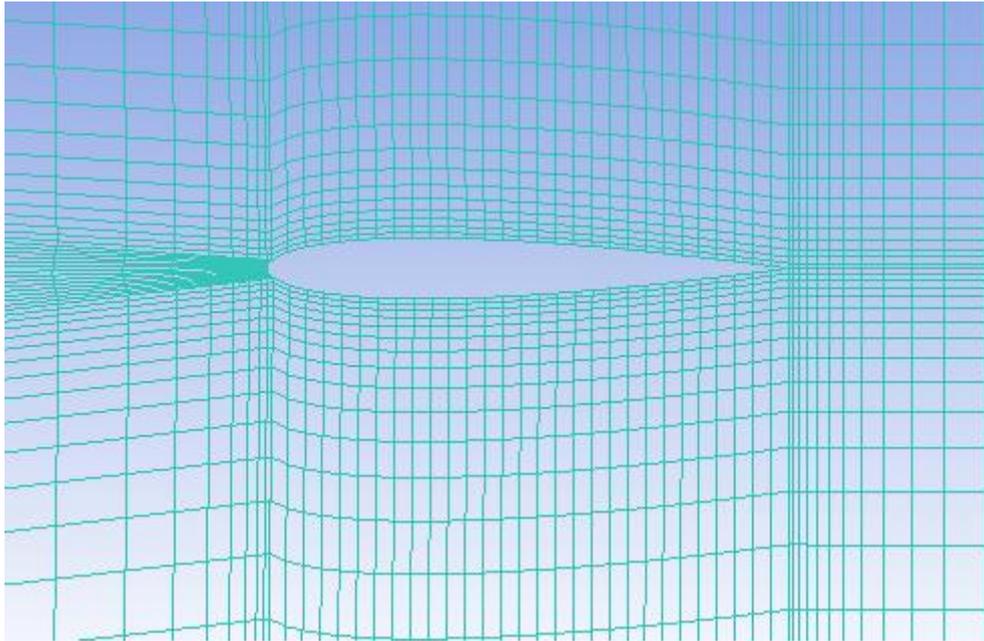
Table 7. Important transport parameters for this case

### 3.4 Geometry and Meshing

A 35 × 30 m rectangular 2-D structured mesh was initially constructed using ANSYS ICEM CFD package with total number of nodes as 4910 as shown in Fig. 9. This mesh gave very poor results while comparing it with the data obtained from [10]. This mesh was considered as a preliminary one to perform a grid independence test. But, any increment in the number of elements would make the simulation unstable for a given time step. A massive decrement in time step was desired which would have made the simulations take too much time in addition to what it was already taking.



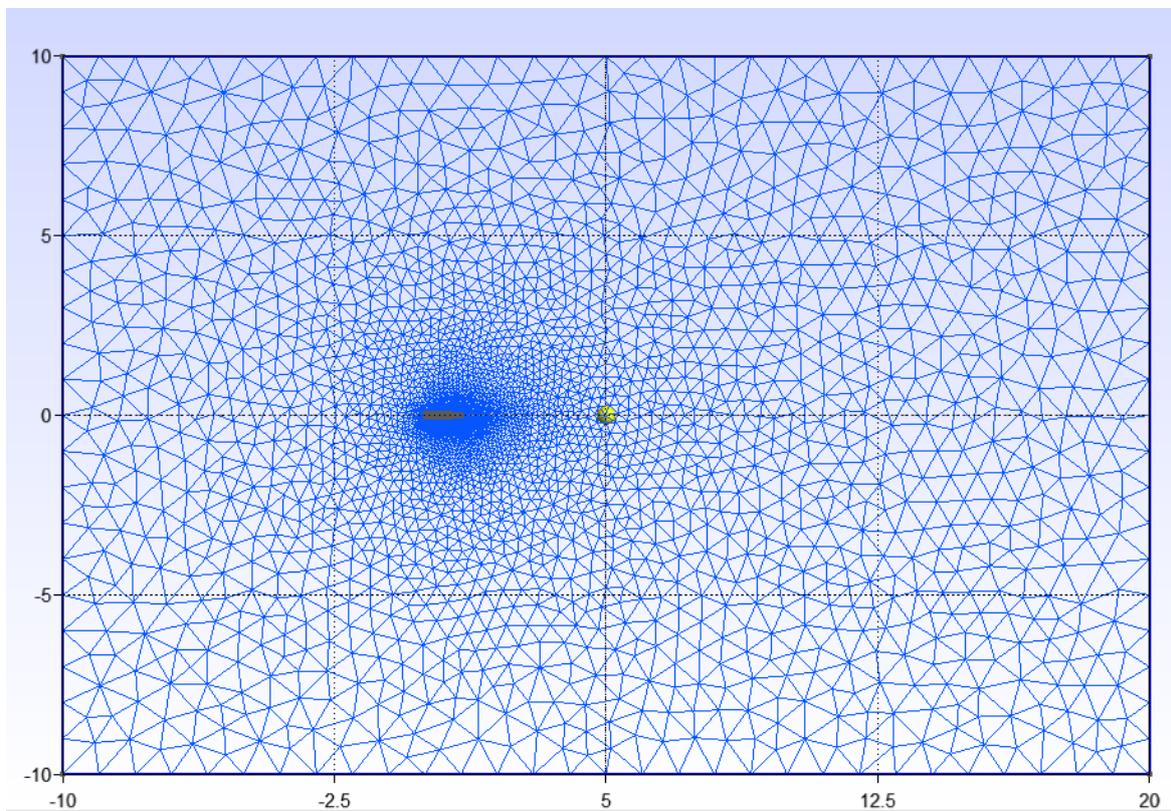
(a)



(b)

**Fig. 9 Preliminary structured mesh made using ICEM**

In an attempt to look for simpler meshing techniques which can provide greater accuracy in reasonable time step, unstructured meshing was adopted using Gmsh (version 4.3.0). A characteristic length of 0.005 was used on the airfoil wall.



**Fig. 10 Unstructured mesh with 12938 cells**

The domain size was 30×20 m and it was extruded in the z-direction by 1 m with 1 cell. The mesh was clustered automatically very well near the wall compared to the previous structured mesh. This promised better results.

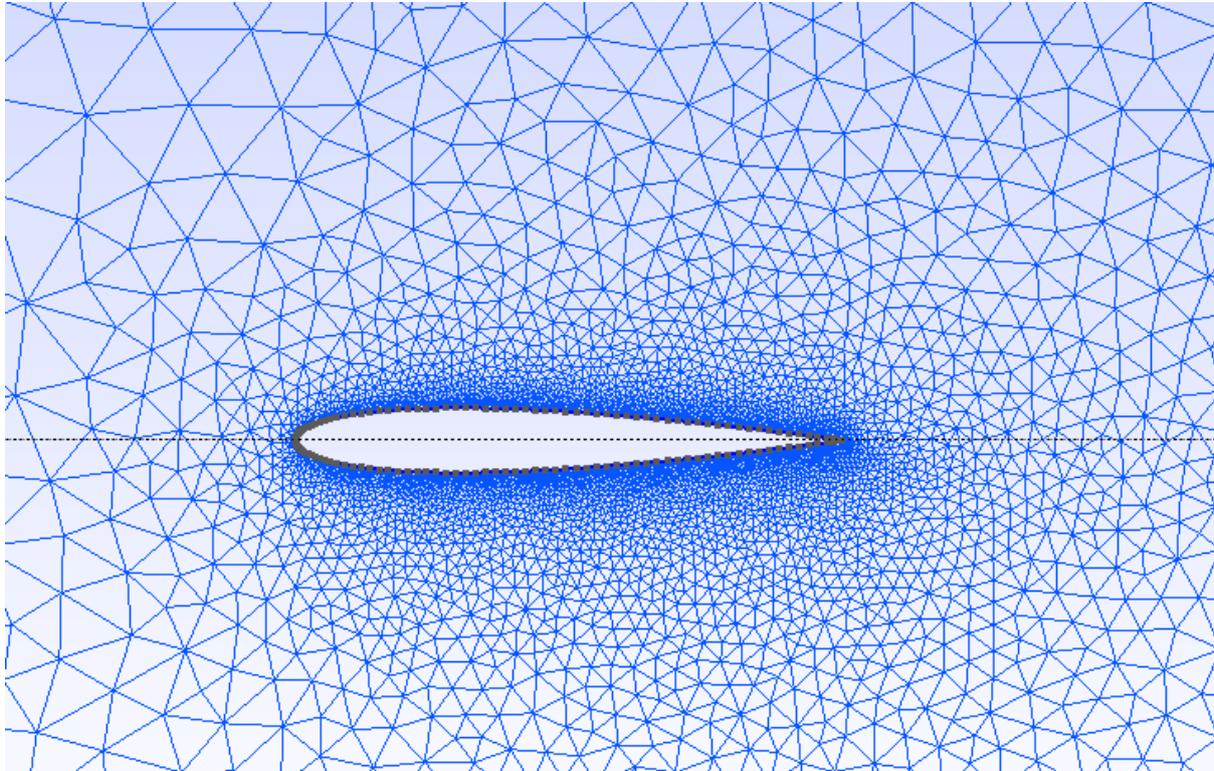


Fig. 11 Automatic mesh refinement in Gmsh

Trying to refine the mesh gave almost similar results to that obtained in the above mesh. Moreover, results were closer to the data as never before. However, after some extent (characteristic length of 0.002 and less), time step had to be reduced by huge amount. Therefore, this mesh was accepted to perform the rest of the simulations.

It is important to note that the mesh generated in gmsh is of advanced version which would not be recognized by *gmshToFoam* command in OpenFOAM – v1812. While exporting the mesh, Version 2 ASCII should be chosen. Before running the command in OpenFOAM, the mesh file needs to be edited – add “\$end” at the end of the file. This will ensure that the mesh is generated successfully.

### 3.4 Boundary conditions and setting up the case files

In Fig. 10, the left boundary is treated as inlet, right boundary as outlet, top and bottom boundaries are named as far-field. Table 8 lists all the boundary conditions set up in the *O* folder in the case directory.

	<b>Inlet</b>	<b>Outlet</b>	<b>airfoil</b>	<b>farfield</b>
<b>U</b>	freestream	freestream	noSlip	freestream
<b>B</b>	zeroGradient	zeroGradient	zeroGradient	fixedValue
<b>nut</b>	freestream	freestream	nutUSpalding WallFunction	freestream
<b>nuTilda</b>	freestream	freestream	fixedValue	freestream
<b>p</b>	freestreamPressure	FreestreamPressure	zeroGradient	freestreamPressure

Table 8. *O* directory

Note:  $pB$  is a correction variable to compensate for error and keep the magnetic flux divergence free, hence no changes required.

The free stream values of  $\nu_{\text{t}}$  and  $\nu_{\text{Tilda}}$  are obtained using the calculations suggested in [9].

Following modifications need to be made in the *system* directory:

- a) *fvSolutions*: Set solver for  $p$  as GAMG with GaussSeidel as a smoother for the same reason mentioned in the previous chapter. Add  $\nu_{\text{Tilda}}$  to the list too and set the smoother for U, B and  $\nu_{\text{Tilda}}$  as GaussSeidel.
- b) *fvSchemes*: The following should be added to the divergence schemes (need to make it second order and solution bounded) to incorporate the turbulence terms and the magnetic transport equation terms (eqn 3):

```
div(phiB,U)    bounded Gauss linearUpwind grad(U);
div(phi,B)    bounded Gauss linearUpwind grad(B);
div(phi,nuTilda)    bounded Gauss linearUpwind grad(nuTilda);
div(phiB,((2*DBU)*B)) Gauss linear;
div((nuEff*dev2(T(grad(U)))) Gauss linear;
```

The closure coefficient varies with change in value of  $Ha/Re$ . Table 9 lists all the values at different  $Ha/Re$  which needs to be added in the *turbulenceProperties* file in the case directory.

<b>B (T)</b>	<b>Ha/Re (*10<sup>3</sup>)</b>	<b>C<sub>v1</sub></b>
1	0.5477	7.185
2	1.095	7.271
4	2.19	7.442

Table 9. Closure coefficients for the Modified Spalart-Allmaras model in this case

In order to obtain the lift and drag coefficients for the airfoil, we need to add the following lines to the *controlDict* file:

```
functions
{
forces
{
type forceCoeffs;
libs ("libforces.so");
outputControl timeStep;
outputInterval 1;
patches
(
airfoil
);
```

```

p p;
U U;
rho rhoInf;
CofR ( 0 0 0 );
liftDir ( 0.4226 0.906 0 );
dragDir ( 0.906 0.4226 0 );
pitchAxis ( 0 0 1 );
magUInf 10;
lRef 0.906;
Aref 0.906;
}
}

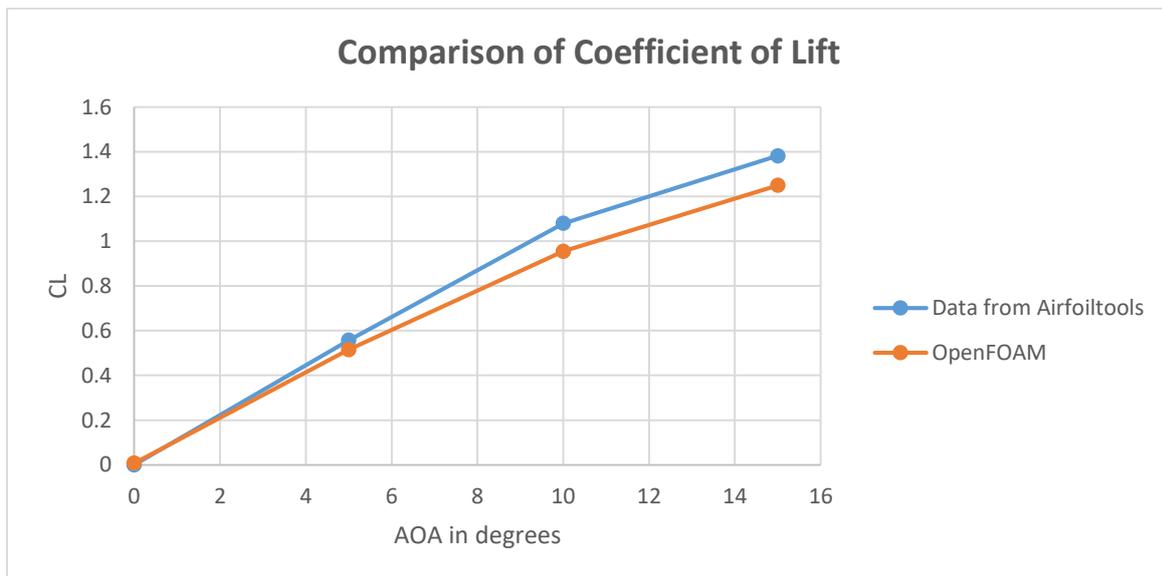
```

This will display the values of the coefficient and the forces while the simulation is running in the display.

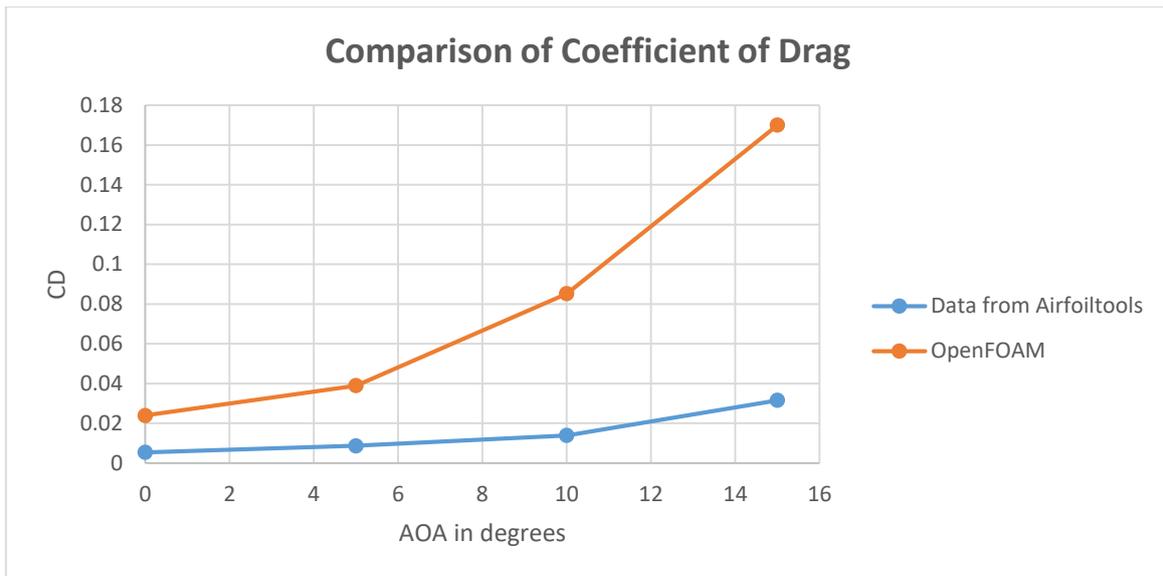
### 3.4 Results

As mentioned before, *mhdTurbFoam* is a transient solver. We need a steady state solution to obtain the  $C_L$  and  $C_D$  value. But one can run the simulations till there is no significant change in these values upto 2 or 3 decimal values. Though this may not give us the exact value, it will be close enough to them to be accepted.

The numerical results without the magnetic field were compared with the experimental data available at [10].



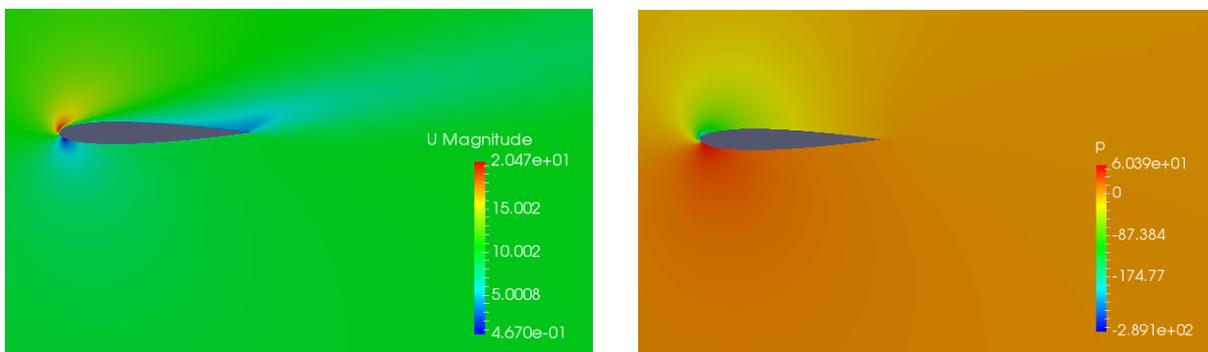
(a)



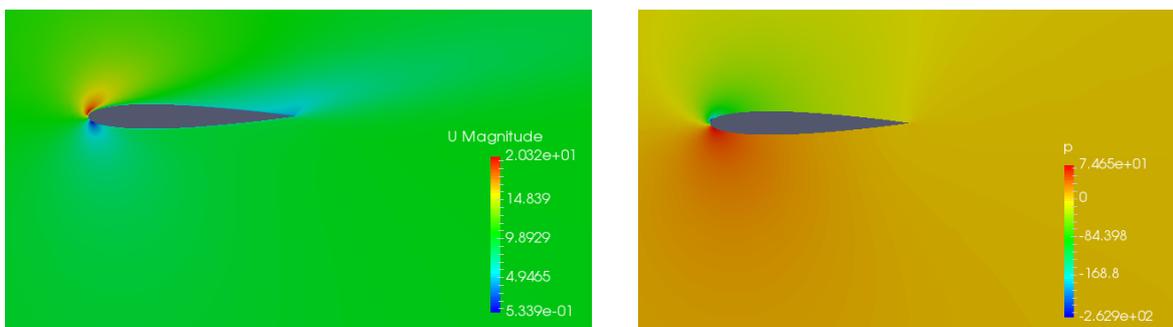
(b)

Fig. 12 Comparison of *mhdTurboFoam* results with data available

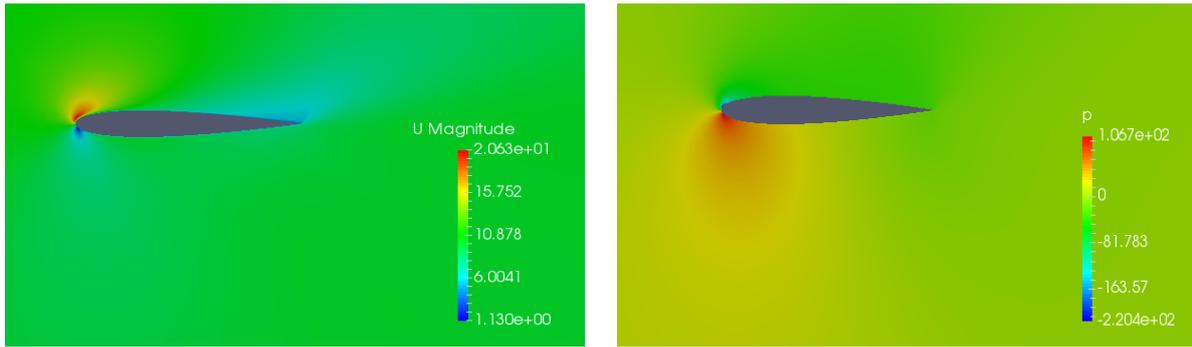
It is observed that the Spalart-Allmaras model over predicts the drag coefficient. This is axiomatic as the model assumes turbulent flow everywhere. However, in reality the flow is laminar to certain extent on the leading edge and then turbulent regime starts. A more accurate prediction of drag can be done by dividing the regions to laminar and turbulent.



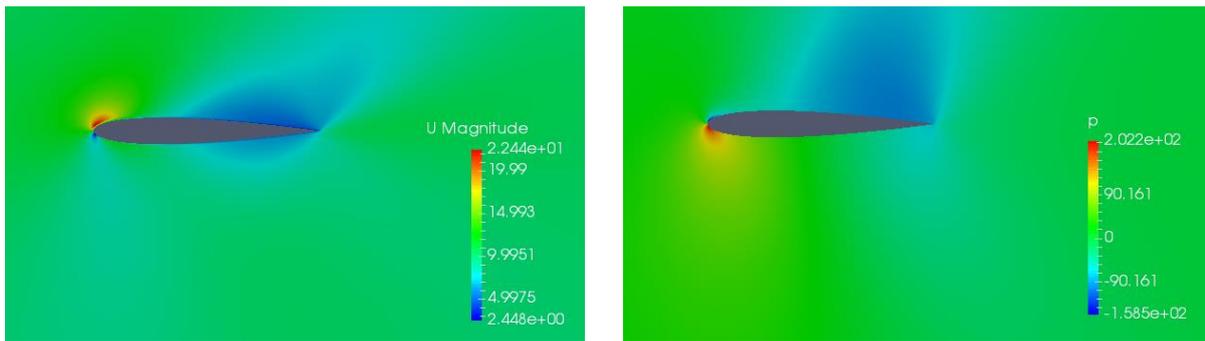
(a)  $B = 0$



(b)  $B = 1 \text{ T}$ ,  $Ha/Re = 0.5477$



(c)  $B = 2 \text{ T}$ ,  $Ha/Re = 1.095$

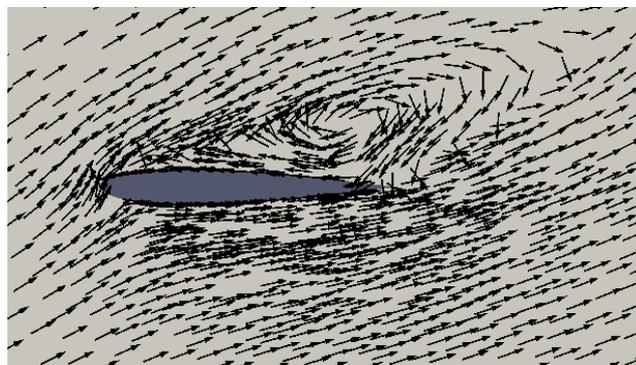


(d)  $B = 4 \text{ T}$ ,  $Ha/Re = 2.19$

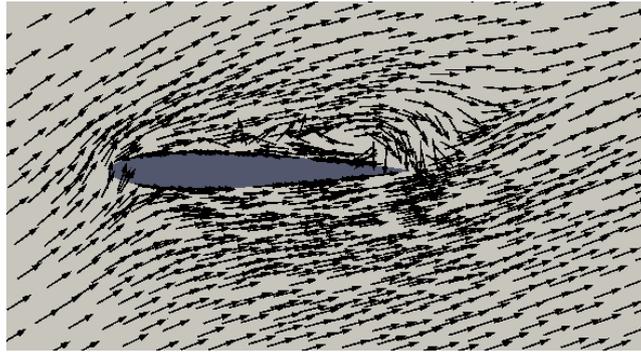
Note: Only few results are shown here for representation purpose. Rest of the contours are available at "All contours.docx" file.

**Fig. 13 Contours of Velocity magnitude and Pressure at  $AOA = 15^\circ$**

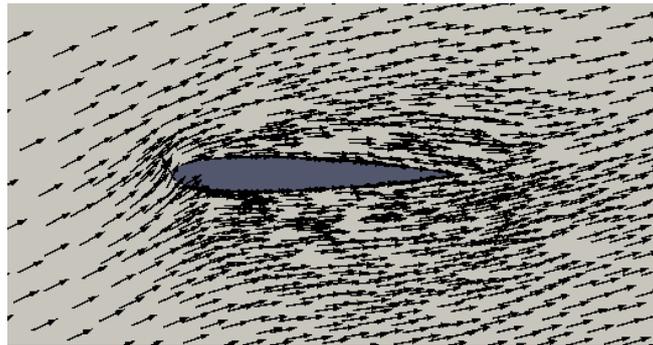
Numerical simulations were performed over a wide range of angles of attack – from  $0^\circ$  to  $25^\circ$ . Fig. 13 shows one set of such simulations. On comparing (a) and (b), we notice how the chord normal, steady magnetic field helps in minimizing the wake region behind the airfoil. As the magnetic field increases, the flow starts to slow down above the surface towards the trailing edge. However, the pressure above the surface of the airfoil decreases with increasing magnetic intensity.



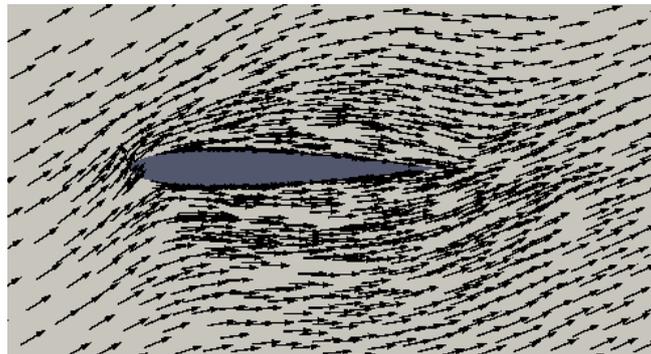
(a)  $B = 0$



(b)  $B = 1 \text{ T}$ ,  $Ha/Re = 0.5477$



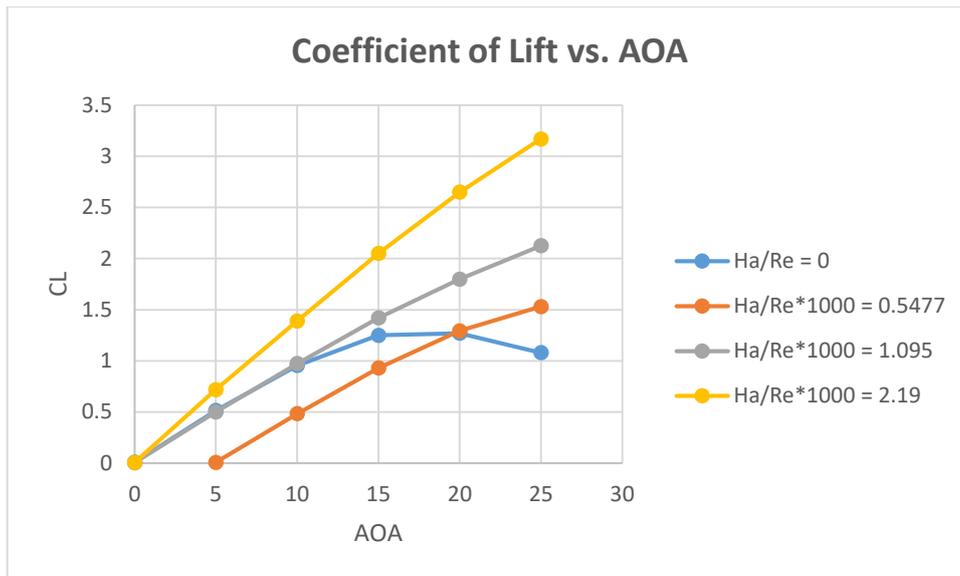
(c)  $B = 2 \text{ T}$ ,  $Ha/Re = 1.095$



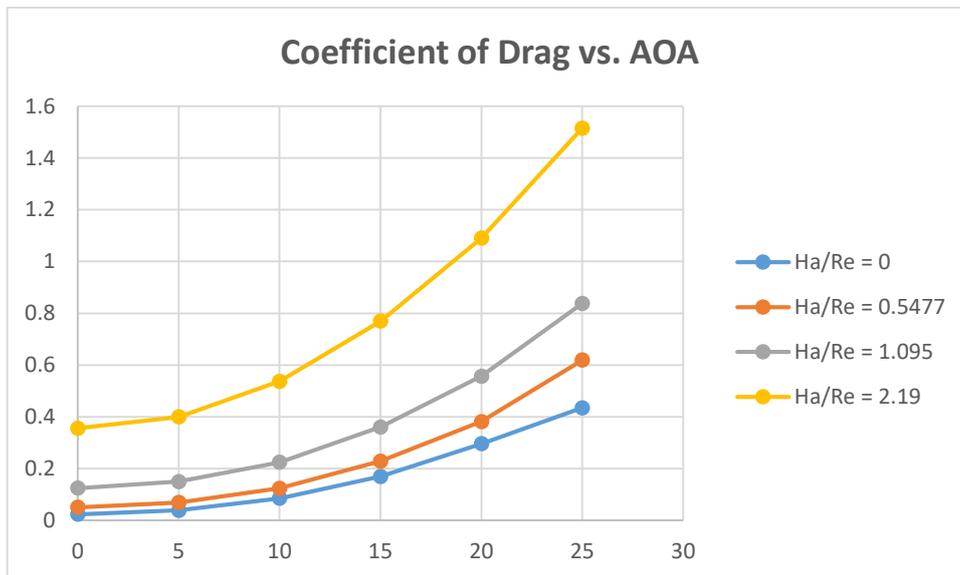
(d)  $B = 4 \text{ T}$ ,  $Ha/Re = 2.19$

Fig. 14 Elimination of turbulent vortices in the flow at  $AOA = 25^\circ$

Fig. 14 shows how increasing magnetic field intensity helps elimination of turbulent vortices and convert the flow from turbulent to laminar. The streamlines were generated using the Glyph tool in Paraview with a scale factor of 0.1 and 5,00,000 sample points.

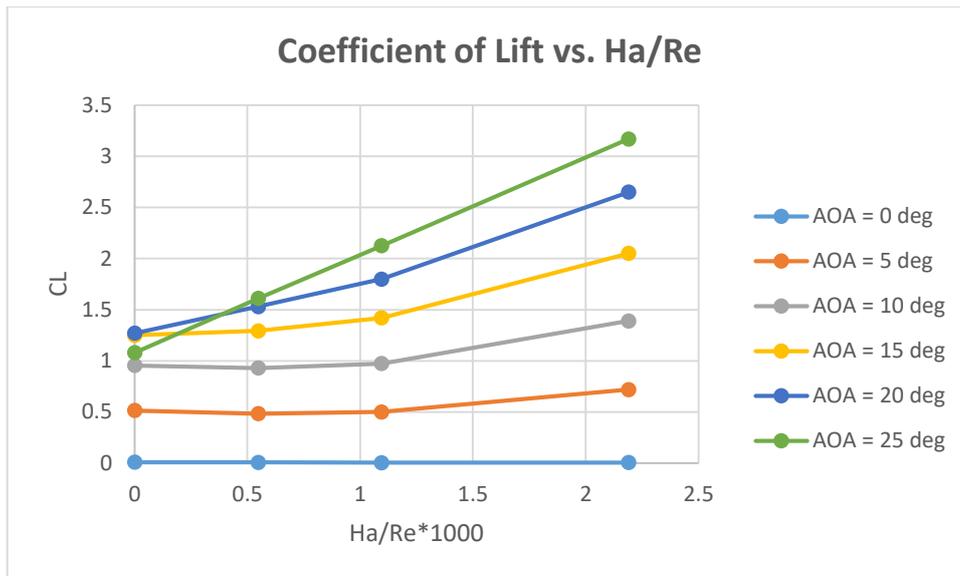


(a)

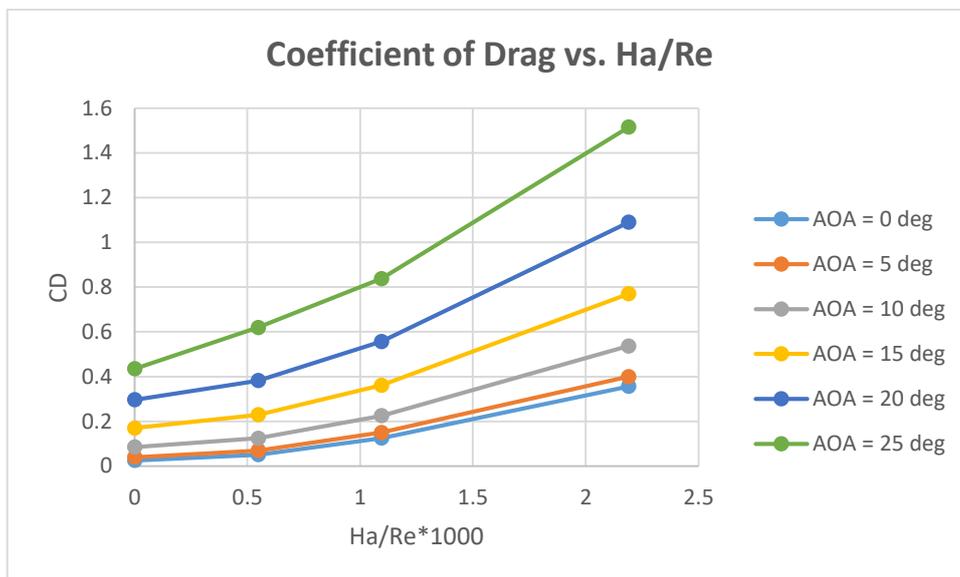


(b)

**Fig. 15 Variation of coefficient of drag and lift at different AOA and at given magnetic field intensity**



(a)



(b)

Fig. 16 Variation of coefficient of drag and lift at different magnetic field intensity and at given AOA.

### 3.4 Conclusion

Based on the results from Fig. 15(a), we see that the effect of increasing magnetic field is that the stall is delayed- the curve is shifted to the right. Further increase in magnetic field intensity seems to increase the lift significantly and also delay the stall. This increase in lift can be accounted to the fact that the pressure over the top surface of the airfoil decreases, which in turn results in increase in suction, thus enhancing the lift (Fig. 16(a)). It is also observed that the coefficient of drag increases with increasing strength of magnetic field.

The amelioration in lift coefficient is more pronounced than drag coefficient. However, as magnetic field increases, the drag coefficient supersedes lift coefficient in terms of percentage increase by more than 100 %. But, this might be a little exaggerated as we know that SA model overestimates drag. In this study, a magnetic field of 4 T increased the drag more than it increased the lift. Nevertheless, this study suggests that a moderate magnetic field strength is desired for economical use of this concept.

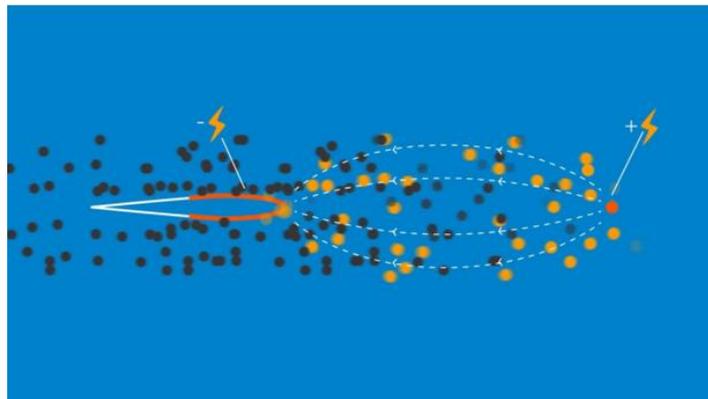


Fig. 17 MIT's solid-state aircraft concept [8]

Fig. 17 shows the ionocraft. The basic principle is that the air is ionized in front of the wing and the ions collide with neutral air molecules, partially transferring their momentum to it. As these neutral molecules are ejected from the ionocraft, they are, in agreement with Newton's Third Law of Motion, equal and opposite forces, so the ionocraft moves in the opposite direction with an equal force. The idea is if we reduce the distance between the released ions and the leading edge, and keep the cathode a little farther away, then this might increase the probability of allowing the ionized air to flow over the airfoil. Then, one can apply magnetic field to control the lift. Though this method might decrease the thrust solely generated by EAD, it would be a hybrid method to propel the same aircraft. This would be an interesting experiment to work out.

Another area where I can see this has application is in space. The increment in lift and drag forces can be taken advantage of by producing a secondary power supply in spacecraft having ion propulsion. This method, though must have been ruled out due to vacuum in space. But, now that we have ionized gas in such space craft its feasibility can be tested.

Yet another area of application is in sea-water where this method can be incorporated on hydrofoils. The main drawback of this is using huge massive magnets which made it infeasible back in the 1960s. But, it has a huge potential as a secondary in these applications instead of being the primary.

Future work includes studying the effect of steady and pulsed wall normal magnetic field in OpenFOAM.

## References

- [1] P.A Davidson, An Introduction to Magnetohydrodynamics, Cambridge University Press, 2001
- [2] S. Smolentsev, Application of the “K–e” model to open channel flows in a magnetic field, International Journal of Engineering Science 40 (2002) 693-711, doi: 10.1016/S0020-7225(01)00088-X
- [3] Jean-Fran-atilde, Dietiker, ois, & Hoffmann, K. A. (2003). *Modified One-Equation Turbulence Models for Turbulent Magnetohydrodynamic Flows*. Journal of Thermophysics and Heat Transfer, 17(4), 509-520. Doi: 10.2514/2.6796
- [4] Brouillette, E. C. (1967). Magneto-Fluid-Mechanic Channel Flow. I. Experiment. Physics of Fluids, 10(5), 995. doi:10.1063/1.1762253
- [5] T. Weier and G. Gerbeth, Control of Flow Separation from a Hydrofoil Using Lorentz Forces
- [6] Mutschke, G., Gerbeth, G., Albrecht, T., & Grundmann, R. (2006). Separation control at hydrofoils using Lorentz forces. European Journal of Mechanics - B/Fluids, 25(2), 137–152. doi:10.1016/j.euromechflu.2005.05.002
- [7] A. Sedaghat, Numerical study on flow separation control over NACA0015 aerofoil using electromagnetic fields, Theoretical and Applied Mechanics Letters, doi: 10.1063/2.1305203
- [8] MIT's electric aircraft initiative (<https://electricaircraft.mit.edu/>)
- [9] Free-stream turbulence boundary conditions ([https://www.cfd-online.com/Wiki/Turbulence\\_free-stream\\_boundary\\_conditions](https://www.cfd-online.com/Wiki/Turbulence_free-stream_boundary_conditions))
- [10] Airfoiltools (<http://airfoiltools.com/airfoil/details?airfoil=n0012-il>)