ISSN 2229-5518

Numerical Simulation of Laminar Flows and Turbulent Flows over Flat Plate Using OpenFOAM

Hoai Nguyen
Department of Mechanical Engineering, College of Technology, University of Danang, Viet Nam
Email: hoainguyen.tme@gmail.com

Abstract— It is well known that OpenFOAM has become a very popular tool for research work in different fields and especially, in computational fluid dynamics. But, it is also known its lack of detailed documentation supporting solvers made using the set of libraries given by OpenFOAM. In this way, it gets to be important to build up suitable confirmations that can be helpful to users. With this reason, executable solvers available in OpenFOAM 3.0.0 version to solve laminar flow and turbulent flow problems are tested. These problems as a basis for the development of numerical methods to solve more complex issues related to the change of basic parameters such as temperature, velocity, pressure, and density of flows. In this work, laminar flow and turbulent flow in a flat plate are simulated through the relationship between aero-dynamics parameters. The results indicate that the relationship between the aero-dynamics parameters depends heavily on the position and shape of a flat plate that flows go through. The physical mechanisms of these phenomena are analyzed, which can be a basis for predicting the structure of dynamic flows in real-world operation.

Index Terms— Numerical Simulation, Turbulent Flow, Incompressible Flow, Finite Volume Method, OpenFOAM, CFD, Laminar Flow

1 Introduction

THE core numerical method was implemented in the open-source OpenFOAM toolbox (hereafter OF). This code was chosen because of several reasons. OF was originally developed as a hiend C++ classes library (Field Operation and Manipulation) for abroad range of fluid dynamics applications and quickly became very popular in industrial engineering as well as in academic research [1]. Within OpenFOAM [2] both approaches are implemented, for instance, the laminar flows based pisoFoam and the turbulent flows based icoFoam. Nevertheless, in spite of many attractive features, the OF toolbox has some disadvantages, as well .The most crucial are: the absolute lack of default settings and the absence of the quality certification and, as a consequence, the absence of high-quality documentation and references. The huge amount of different numerical schemes, algorithms and mathematical models creates the illusion that any problem can be solved. Actually, the available catalogs of mathematical models are not perfect and many of them are subject to further research. Moreover, the acceptability of mathematical models for solving complex (multi-physics) problems has yet to be analyzed. The limits for application of most of the models are also not clearly understood.

An OpenFOAM simulation is characterized by a group of subdirectories, each containing specific files, as shown in Fig.1. The file structure of an OpenFOAM case is composed of a system directory, where parameters connected with the solution procedure defined a constant directory which contains mesh information and physical properties for case; and the time directories, where initial/boundary conditions and results for each recorded time step are saved [3].

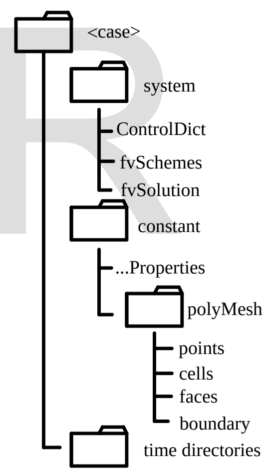


Fig.1. OpenFOAM case directory structure

2 K-EPSILON TURBULENCE MODEL

This is the most common model used in Computational Fluid Dynamics to simulate mean flow characteristics for turbulent flow conditions. It is two equation model which gives a general description of turbulence by means of two transport equations. The original impetus for the K-epsilon model was to enhance the mixing-length model, and also to find an alternative to algebraically prescribing turbulent length scales in moderate to high complexity flows.

Not at all like prior turbulence models, k-ε model concentrates on the mechanisms that influence the turbulent kinetic energy. The mixing length model lacks this kind of generality. The underlying assumption of this model is that the turbulent viscosity is isotropic, in other words, the ratio between Reynolds stress and mean rate of deformations is the same in all directions.

For turbulent kinetic energy k and dissipation ε are described in the following equation:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial(\rho k u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\frac{\mu_t}{\delta_k}\frac{\partial k}{\partial x_j}\right]+2\mu_t E_{ij}E_{ij}-\rho\varepsilon \qquad (1.1)$$

$$\frac{\partial(\rho \varepsilon)}{\partial t}+\frac{\partial(\rho \varepsilon u_i)}{\partial x_i}=\frac{\partial}{\partial x_j}\left[\frac{\mu_t}{\delta_\varepsilon}\frac{\partial \varepsilon}{\partial x_j}\right] \qquad (1.2)$$

$$+C_{1\varepsilon}\frac{\varepsilon}{k}2\mu_t E_{ij}E_{ij}-C_{1\varepsilon}\rho\frac{\varepsilon^2}{k}$$

In the derivation of the standard k−ε model the flow is assumed to be fully turbulent and the effects of molecular viscosity to be negligible. Therefore the standard k−ε model is a high Reynolds number turbulence model valid only for fully turbulent free shear flows that cannot be integrated all the way to the wall [4]. A turbulence model that can be integrated all the way to the wall is denoted in the literature by a low Reynolds number turbulence model or a low Reynolds number version. Several so called low Reynolds number k−ε models have been proposed over the years [5].

In OpenFOAM, k−ε model is located constant directory as shown in Fig.2

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|  \\    /   O peration     | Version:  1.6                                   |
|   \\  /    A nd           | Web:      www.OpenFOAM.org                      |
|    \\/     M anipulation  |                                                 |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      RASProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

RASModel        kEpsilon;

turbulence      on;

printCoeffs     on;


// ************************************************************************* //
```

Fig.3. Turbulence model properties

3 NUMERICAL METHOD

The finite volume method(FVM) is used in OpenFOAM to obtain a numerical solution for flow problems. In FVM, the solution to the partical differential equations that describe the flow behavior is approximated by subdividing the computational domain into a finite number of control volume elements and applying conservation laws to each of them.

• Hoai Nguyen is currently teaching in Department of Mechanical Engineering, College of Technology, The University of Danang, Viet Nam, PH: +84-934566516. E-mail: hoainguyen.tme@gmail.com

The process of subdividing continue into finite, or discrete, quantities is know as dicretication[6].

After a finite number of equations describing conservation laws are generated, they must be solved to find the values of the variables of interest for a given flow [7]. Due to the nature of the partial differential equations that describe the fluid's behavior, a set of non-linear coupled equations is usually obtained. These complications make obtaining a solution to the system impossible unless iterative solution methods are employed. The movement of the typical turbulent flow is described in Figure 2.

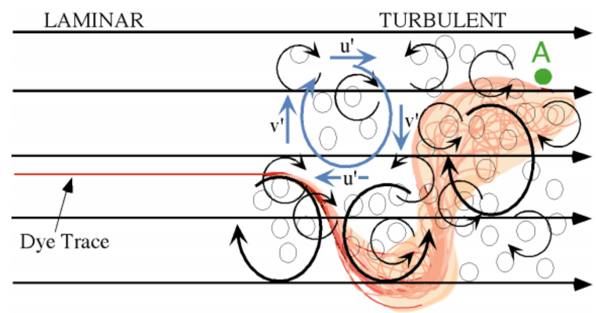


Fig.2. Tracer transport in laminar and turbulent flow [8]

3.1. Discretization

In the OpenFOAM simulations, a pseudo-time is introduced with two reasons: firstly, to control the amount of iterations performed by the solver and second, to specify the frequency of the output of the solution to the computers hard disk. The controlDict dictionary of OpenFOAM is used to realize these functions. To achieve results through the definition of the values of the endTime, deltaT, writeControl, and writeInterval options in controlDict.

3.2. Equation Discretization

In OpenFOAM, an approximate components of the conservation and turbulence model equations are specified through the fvSchemes dictionary found in the case's system directory.

The flow cases studied are steady, so the temporal derivatives in all equations are not taken into account [9]. The second-order Gaussian integration scheme is utilized for every term in momentum and turbulence model equations that includes a derivative. Since OpenFOAM calculates values at each element's center, values must be interpolated from cell to face centers. The central difference interpolation is utilized for all gradient terms. Upwind differencing scheme [10] is employed for convective terms in all equations. The second-order scheme is used for the terms in the momentum equations and the first-order scheme is applied to the turbulence transport equations. The central difference interpolation scheme is used for the diffusion coefficient in all diffusive terms, and an explicit secondorder non-orthogonal correction method is employed for surface-normal gradients.

3.3. Solution Method

In this paper, icoFoam and pisoFoam solves the incompres-

ISSN 2229-5518

sible laminar and turbulence Navier-Stokes equations using the PISO (Pressure Implicit with Splitting of Operators) algorithm [11] which is an efficient method to solve the Navier-Stokes equations in unsteady problems.

The algorithm can be summed up as follows:

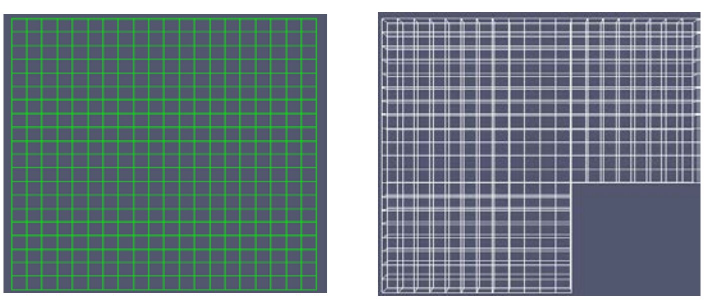
1. Set the boundary conditions.
2. Solve the discretized momentum equation to compute an intermediate velocity field.
3. Compute the mass fluxes at the cells faces.
4. Solve the pressure equation.
5. Correct the mass fluxes at the cell faces.
6. Correct the velocities on the basis of the new pressure field.
7. Update the boundary conditions.
8. Repeat from 3 for the prescribed number of times.

Increase the time step and repeat from 1.

4 COMPUTATIONAL DOMAIN

The geometry for two flow cases will be discussed in this section. A description of the computational domain for the two-dimensional flow over a flat plate with zero pressure gradient will be presented. To model the flow cases as 2-D in OpenFOAM, the two-dimensional grid must be extended one unit in the third dimension, which creates extra domain boundaries.

The meshes used in this study are nested, meaning that each coarser grid is exactly every-other-point of the finer grid. The naming convention for the meshes consists of the amount of nodes in the x- and y- directions. Note that the computational domains are not defined in terms of meters, or feet, but in terms of dimensionless units. The meshes are created by using blockMesh in OpenFOAM as shown in Fig.4a,b



(a) Turbulence flow case (b) Laminar flow case

Fig.4. Mesh simulation in OpenFOAM

5 BOUNDARY CONDITIONS

In an OpenFOAM simulation, the boundary and initial conditions for eachflow v ariable are specified in the case's 0 time directory. A fixed velocity value of 1 m/s, is used as the inlet boundary condition. For the pressure at the inlet and plate boundaries, OpenFOAM's Neumann-type boundary condition, known as zeroGradient, is assigned. A no-slip boundary condition is used on the adiabatic plate surface for the velocity. The outlet is assigned the zeroGradient condition for the velocity and OpenFOAM's Dirichlet-type boundary condition, fixedValue, for pressure (1 atm). The zeroGradient boundary condition is assigned as to the top boundary for all variables. Temperature boundary conditions are not necessary because the simulation is run as incompressible.

6 RESULTS AND DISCUSSION

6.1 Aerodynamic stream simulation

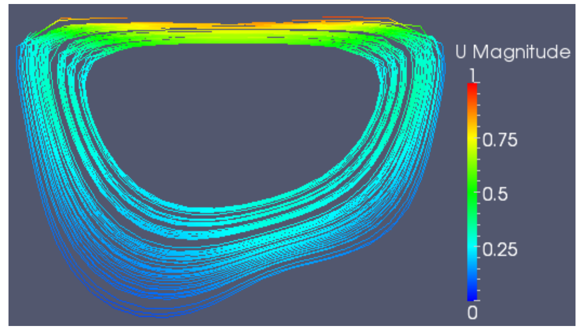


Fig.5. Aerodynamic stream simulation of laminar flow

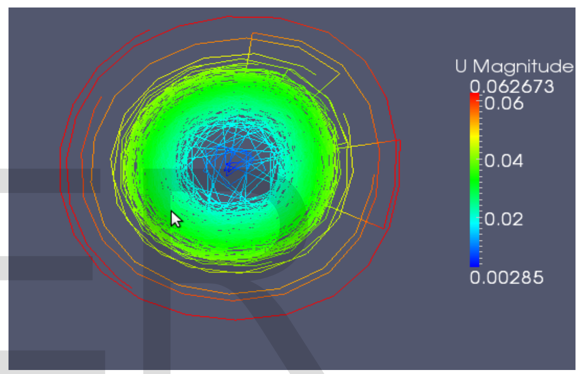


Fig.6. Aerodynamic stream simulation of turbulence flow

Laminar flow could be depicted as the flow of a fluid whenever each and every particle belonging to the fluid is a follower of a consistent course, routes which usually under no circumstances obstruct with each other. One result of laminar movement would be that the speed belonging to the fluid is actually constant at any time inside fluid whereas on the other hand turbulent flow could be depicted as the uneven, unfrequented movement of fluid which is viewed as a little whirlpool regions. The speed of such a liquid is certainly not really steady at every point. The results are shown in Fig.5-6.

6.2 Velocity of turbulence flow and laminar flow over flat plate

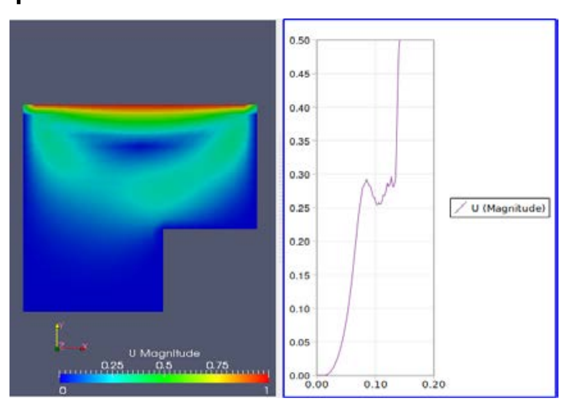


Fig.7. Velocity of laminar flow over flat plate

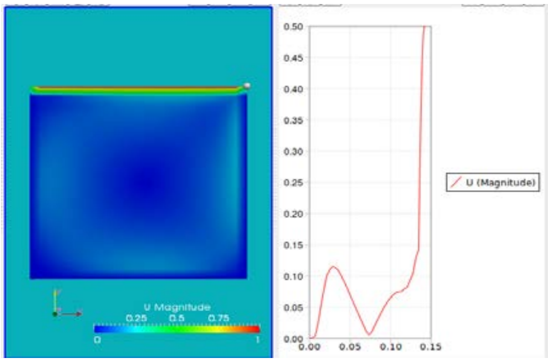


Fig.8. Velocity of turbulence flow over flat plate

The velocity of the laminar flow is variable gradually increase trend as Fig.7. Additionally, it depends on speed of the top wall, and it reaches the maximum value (1 m/s) at the edge of the top wall. The closer to the top wall area, the change in flow velocity occurs more. In other areas of the flat plate, the velocity hardly changed.

Similarly, the velocity of the turbulence flow has changed from 0 value to values near velocity of the top wall. This arises because of viscosity, υ, which is a fluid's resistance to flowing.

6.3 Pressure of turbulence flow and laminar flow over flat plate

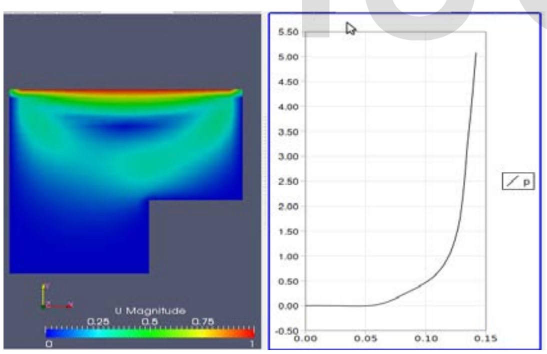


Fig.9. Pressure of laminar flow over flat plate

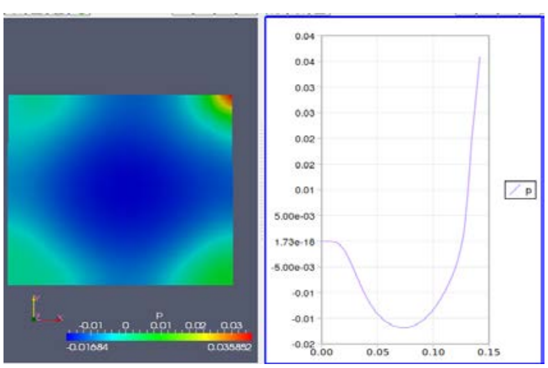


Fig.10. Pressure of turbulence flow over flat plate

The results in Fig.9-10 demonstrate that pressure of laminar flow and turbulent flow depends on its velocity. However, the flow rate is proportional to the square root of the pressure gradient, whereas in laminar flow, flow rate is directly proportional to the pressure gradient. Therefore, the shape of the graphs is the differ. Changing pressure range of laminar flow is narrower than the turbulence flow.

6.4 Turbulence kinetic energy k

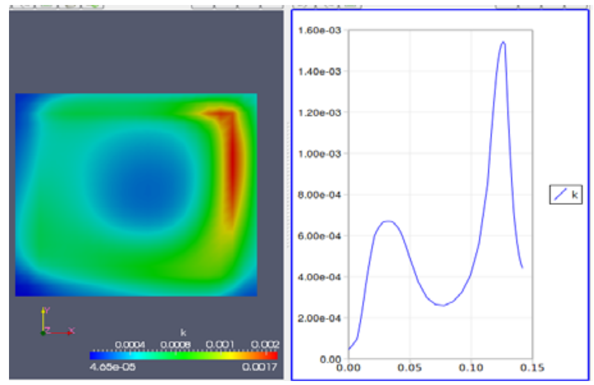


Fig.11. Turbulence kinetic energy k simulation

Turbulence kinetic energy graph is similar to velocity of turbulence flow. This is explained by the turbulence kinetic energy depends on the average velocity flow under different directions (Ux , Uy , Uz), so it also tends to increase according to boost velocity of turbulence flow. Especially, it is determined near the border of the top wall and the right side wall. Additionally, change the value of the kinetic energy generated extremes tangled clearly on the graph.

6.5 Turbulence dissipation rate ε

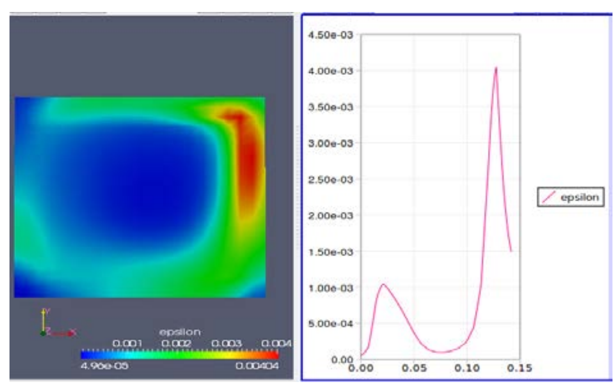


Fig.12. Turbulence dissipation rate ε simulation

Essentially, the turbulence dissipation rate is the turbulence kinetic energy is converted into thermal internal energy. Therefore, the shape of turbulence dissipation rate losely resembles dissipate turbulence kinetic energy graph. However, the range of the turbulence dissipation rate is greater than turbulence kinetic energy . This is due to the increased Reynold, viscous stresses different impact on the parameters (k and ε).

7 CONCLUSION

OpenFOAM is an open source code and like most of them is slightly more complicate to use it than commercial codes. It requires some Linux and C++ knowledge which it makes a little harder the beginning for a new user. Nonetheless, since the code is open the user is permited to change it and adjust it to his/her own necessities which makes it a very interesting tool for the future.

Computational fluid dynamics simulations were performed with OpenFOAM for two different flow cases by k- ε model. Results achieved indicate that the parameters: velocity, pressure, turbulence kinetic energy, turbulence dissipation rate increase rapidly when it is near the border the top position.

For the turbulence flow over flat plate, it is expected to find the maximum velocity close to the surface. In fact, this is key point for study the velocity profiles.

Additionally, the capability of OpenFOAM to simulate the flow physics for all the configurations with reasonable accuracy without any additional customizations is clearly demonstrated by comparing the predictions with computational data. Though some properties of laminar flow and turbulence flow could be studied from the experiments, the numerical calculations can be carried out with very small time steps to predit the structure of dynamic flows in real-world operation.

8 FUTURE WORK

If the study is to be proceeded there are some aspects that can be improved or more deeply studied.

Firstly, there is always the opportunity to refine the mesh so that the solution is more precise, and in this case where there are different phases, the interface between them would be more strongly characterized if the size of the cells is smaller.

Another improvement that can be made is to change from 2-Dimensional to 3-Dimensional simulations. It is always more realistic to make a simulation in 3-D, because making some assumptions to simplify the case to a 2D can make the case lose accuracy in the results, particularly when they are to be contrasted with experimental results.

Here, like in most of the simulations, a turbulence model is adopted. However, if someone wants to directly solve the turbulence equations without any approximation the Direct Numerical Simulation can be used. In the Direct Numerical Simulation there is no approximation or averaging and they can solve three dimensional dynamic and time dependent Navies Stokes equations with the assistance of a supercomputer.

ACKNOWLEDGMENT

The authors are grateful to College of Engineering, The University of Danang, Viet Nam and Chinese Culture University, Taiwan for having made this work possible. The authors also appreciate the technical support of Prof. Tsing-Tshih Tsung, Department of Mechanical Engineering, Chinese Culture University, Taiwan.

REFERENCES

[1] Dmitry A. Lysenko, Ivar S. Ertesvåg, Kjell E. Rian, " Modeling of turbulent separated flows using OpenFOAM", *Computers and Fluids,* vol.80, pp. 408-422, 2013.

[2] OpenCFD Ltd, "The opensource CFD toolbox", http://www.openfoam.com/about/

[3] Hoai Nguyen, Le Van Nguyen, Thanh Phong Pham, "Computation and Testing Oblique Shock over Wedge by Using OpenFOAM", *Proc. 5th World Conference on Applied Sciences, Engineering & Technology*, pp. 261-265, 2016.

[4] Yunho Jang, "Boundary layer analisis with Navier-Stocks equation in 2D channel fow", *Proc. American Society of Mechanical Engineers*, pp.01-05.

[5] Jakirlic, S., & Hanjalic, K., "A new approach to modelling near-wall turbulence energy and stress dissipation", *Journal of fluid mechanics,* vol. 459, pp.139-166, 2002.

[6] Moukalled, Fadl, Luca Mangani, and Marwan Darwish, *The Finite Volume Method in Computational Fluid Dynamics: An Advanced Introduction with OpenFOAM® and Matlab*, vol.113, 2015.

[7] Sebastian Gomez, "Verification of statistical turbulence models in aerodynamic flows", Master dissertation, Dept. of Mechanical Eng., The University of New Mexico, Albuquerque, New Mexico, 2014.

[8] Vidal Garcia, Omar, "Determination of the correlation between turbulence intensity and acoustic noise level-two clockwise-turning rotors case", Master dissertation, Dept. of Mechanical Eng., Lodz University of Technology, Poland, 2011.

[9] Furbo, Eric, Janne Harju, and Henric Nilsson, "Evaluation of turbulence models for prediction of flow separation at a smooth surface", *Report in Scientific Computing Advanced Course-Project* 9, 2009.

[10] Audusse, Emmanuel, and Marie-Odile Bristeau, "A well-balanced positivity preserving "second-order" scheme for shallow water flows on unstructured meshes", *Journal of Computational Physics,* vol.206, pp. 311-333, 2005.

[11] Patankar, S. V. and Spalding, D.B., "A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows", *Int. J. of Heat and Mass Transfer*, vol. 15, Issue 10, pp.1787-1806, 1972.