27th ABCM International Congress of Mechanical Engineering
December 04-08, 2023, Florianópolis-SC, Brazil

COB-2023-0151
INVESTIGATION OF THE INFLUENCE OF GROUND ROUGHNESS ON AIRFLOW IN THE ATMOSPHERIC BOUNDARY LAYER USING OPENFOAM

Gilberto Augusto Amado Moreira
Arthur Leite Guilherme
Patrícia Rafaela Alves Santos
Fabiano Cordeiro Cavalcnati
Universidade Federal da Paraíba, Cidade Universitária - João Pessoa - PB - Brasil - CEP: 58051-970.
gilberto@cear.ufpb.br

Abstract. *This study used the OpenFOAM numerical simulation software to investigate the influence of ground roughness on airflow in the atmospheric boundary layer. Numerical simulation is a technique used to solve complex problems that cannot be solved analytically. Computational Fluid Dynamics (CFD) is a branch of numerical simulation that focuses on solving fluid flow problems. OpenFOAM is an open-source CFD software that allows users to develop their own simulation models and apply them to a wide variety of fluid flow problems. The software is widely used in industry and academia to simulate engineering problems and conduct research in various areas such as aerospace, automotive, environmental, among others. The objective of this study is to evaluate the influence of ground roughness on airflow in the atmospheric boundary layer, which is the region of the atmosphere in contact with the Earth's surface and where the main heat, momentum, and mass transfer processes occur between the surface and the atmosphere. Ground roughness is an important factor in the atmospheric boundary layer as it affects the wind speed and direction, as well as the pressure and temperature distribution on the surface. To perform the simulation, experimental data from an open field with different ground roughnesses were used. The simulation results were validated with experimental data available in the literature. A comparative analysis of velocity, pressure, and temperature distributions for different ground roughnesses was performed. The simulation results showed that ground roughness has a significant effect on airflow in the atmospheric boundary layer. In particular, it was observed that an increase in ground roughness resulted in a reduction in wind speed in the boundary layer. Additionally, the pressure and temperature distribution on the surface were also affected by ground roughness. In summary, this study demonstrated that the use of OpenFOAM software is a valuable tool for investigating airflow in the atmospheric boundary layer. The results obtained can be used to improve the understanding of the physical processes that occur on the Earth's surface and to improve climate and weather prediction. Additionally, this study may provide useful information for the design and operation of structures and equipment exposed to wind and temperature on the Earth's surface.*
Keywords: *Soil Roughness, Atmospheric Boundary Layer, OpenFOAM, Numerical Model.*

1. INTRODUCTION

Soil roughness is an important factor that influences airflow in the atmospheric boundary layer (ABL). This layer is characterized by significant changes in air properties, such as temperature, humidity, wind speed, and pollutant concentration, which are influenced by processes occurring on the Earth's surface, such as evaporation, solar radiation, vegetation, topography, and human activities, which compose the soil roughness. Soil roughness is an important factor that affects the speed, direction, and turbulence of airflow within the ABL (Barasa, Maulidi; Xuemin, Li; Zhang, Yi; et al, 2022). To better understand how soil roughness affects airflow in the ABL, many studies have been conducted using various simulation techniques.

One of the simulation techniques used to study the influence of soil roughness on airflow in the ABL is OpenFOAM (Open Field Operation and Manipulation), a free computational fluid dynamics (CFD) software that allows simulation of fluid flows and heat transfer in complex systems. It is developed by the OpenFOAM Foundation and distributed under the GPL (General Public License).

The main programming language of OpenFOAM is C++, a widely used object-oriented programming language in scientific and engineering software development. In addition, OpenFOAM also has interfaces to other programming languages, such as Python, which can be used to create scripts and automate tasks in simulations (Abbasi, Ali; Annor, Frank Ohene; Van De Giesen, Nick, 2018).

This software is capable of solving a wide range of fluid flow problems, including incompressible, compressible, multiphase, and turbulent flow problems. It uses the finite volume method to discretize the conservation equations of mass, momentum, and energy, and can be run on parallel computers to accelerate the simulation of complex problems (Barasa, Maulidi; Xuemin, Li; Zhang, Yi; et al, 2022).

Furthermore, it is used in various areas, including aerospace, automotive, oil and gas, renewable energy, chemical processes, and medical devices, among others. It is a powerful tool for designing and optimizing systems and processes involving fluid flow and heat transfer.

Studies conducted with OpenFOAM have shown that soil roughness significantly affects airflow (Ahmed Kabir, Ijaz Fazil Syed; Kanagalingam, Sivamoorthy; Safiyullah, Ferozkhan, 2017). The height of soil roughness, the shape and distribution of objects on the soil surface, wind speed, and other environmental conditions affect the turbulence of airflow within the ABL. These studies have provided important information to improve climate prediction and decision-making in many areas, including agriculture, construction, aviation, and wind energy (Cao, Yong; Tao, Tao; Shi, Yujiang; et al., 2023).

Besides studying the influence of soil roughness on airflow in the ABL, OpenFOAM can also be used to study other phenomena related to fluid dynamics, such as vehicle aerodynamics, fluid flow in pipelines, and fuel combustion. OpenFOAM is a powerful and versatile tool that can be used in many different areas of engineering and applied sciences (Qwist, Jesper Roland Kjærgaard; Christensen, Erik Damgaard, 2023).

In summary, the influence of soil roughness on airflow in the atmospheric boundary layer is an important and complex topic, which has been studied using various simulation techniques, including OpenFOAM. It is a powerful and versatile tool that can be used in many areas of engineering and applied sciences, becoming a valuable tool for studying fluid dynamics.

Thus, this article presents an investigation of the influence of soil roughness on airflow in the atmospheric boundary layer (ABL), using the computational fluid dynamics software OpenFOAM. The overall objective of this study is to understand how soil roughness affects the speed, direction, and turbulence of airflow in the ABL. To achieve this goal, we tested some two-equation turbulence models for the airflow in the atmospheric boundary layer over a 2D elevation in the extreme situation of a triangular ridge. Here, the size of the recirculation profiles and the velocity downstream of the ridge are analyzed. The OpenFOAM library was used in this investigation.

2. THEORETICAL FOUNDATION

2.1 Contextualization of the model used

In Juretic and Kozmar (2013) used wind tunnel geometry with von Neumann boundary conditions applied at the inlet and the outlet boundaries to obtain homogeneous mean velocity, turbulent kinetic energy and Reynolds shear stress profiles with various RANS turbulence models, which obtained interesting results and reported difficulties in modeling near the wall due to the shear stress approximation and the experimental results. The standard k-ε model, established by Launder and Sharma (1974), is one of the most common turbulence models, despite being imprecise under conditions of large adverse pressure gradients (Wilcox, 2006). Three other models derived from this are: the model proposed by Hargreaves and Wright (2007), here referred as modified k-ε, which has a modified constant; the k-ε realizable model (Liou et al., 1995) and the RNG k-ε model (Yakhot et al., 1992). Other models of two equations are based on the turbulence specific dissipation rate ω, these are k-ω model (Wilcox, 2006) and the kOmegaSST model, implemented in OpenFOAM as a variant of the SST k-ω model of Menter et al. (2003), beyond the SSG Reynolds Stress turbulence model, proposed by Speziale et al. (1991).

2.2 Physical problem

In general, the evaluation of the turbulence model applied here, which will later be applied to a type of atmospheric circulation, may comprise two main processes that dominate its evolution: a) Mechanical effects caused by the topographic or structural obstacles with which the flow interacts, leading to acceleration or recirculation zones. b) Buoyancy effects caused by the heat flux from or towards the surface, which can generate important vertical air flow accelerations. These convective effects depend on the stability of the atmosphere, so that stratification must be taken into account. Both processes interact with each other and are directly influenced by the intense turbulence that characterizes many environmental flows. However, we focus only on the Mechanic to reproduce Arya and Shipman (1981) work, but to apply in Atmosphere Boundary Layer (ABL) it is necessary to implement the thrust that interacts with the mechanic and can influence the turbulent scales.

2.3 Solver - OpenFOAM library

OpenFOAM is a set of C++ libraries developed to solve complex problems related to fluid mechanics. It was created with the aim of providing a more efficient numerical platform than Fortran, utilizing the new functionality of C++ object-oriented programming. OpenFOAM is free to distribute and allows users to develop specific solvers, which can be integrated with existing tools. It uses finite volume formulation, just like most of the CFD codes currently used in engineering. The libraries of OpenFOAM contain turbulence models (RANS and LES), thermophysical models, radiation models, and wall functions, which can be accessed when developing a solver. Additionally, OpenFOAM offers the capability of integrating complex geometries in calculations through the use of the snappyHexMesh tool. Detailed information regarding the numerical methods used in the code and the available tools can be found in the OpenCFD (2016) documentation.

OpenFOAM has been applied to the study of atmospheric flows in numerous examples. For instance, Garcia and Boulanger (2006) used OpenFOAM to simulate the wind flow over Mount Saint Helens in the United States, while Hussein and El-Shishiny (2009) studied the wind flow over the Giza Plateau in Egypt. Larsson et al. (2009) utilized OpenFOAM to model non-buoyant wind flow over complex terrain, using RANS turbulence models with wall functions to predict wind power production accurately. Churchfield (2010) developed an OpenFOAM solver using LES to simulate buoyant flows under the Boussinesq approximation, focusing on the interaction of the flow with wind turbines. Flores et al. (2014) applied a DES approach implemented in OpenFOAM to understand the buoyant effect in pollutant dispersion in Chuquicamata, a large open-pit mine in Chile.

Our research goal is to compare the same model applied in Ansys CFXTM with OpenFOAM. We aim to study the problem by evaluating turbulence and complex geometry to analyze real topography in the future, even without explicitly considering density variables and the treatment of stratification and buoyancy consequences. This approach can be applied to other models, and our research results will be useful for comparison between Ansys CFXTM and OpenFOAM.

2.4 Governing equations and implementation in OpenFOAM

OpenFOAM uses the finite volume method for numerical representation of the equations governing fluid motion and the message passing interface (MPI) method for parallel computing. The toolbox features a range of numerical schemes, methods and turbulence models. The available turbulence models range from Reynolds averaged Navier–Stokes (RANS) to hybrid RANS/LES (HRL) to LES. It is also possible to resolve all scales using direct numerical simulation (DNS). The governing equations for incompressible fluid flow are the Navier–Stokes equations:

$$\nabla \bullet \vec{U} = 0 \tag{1}$$

$$\frac{\partial \vec{U}}{\partial t} + \left(\nabla \bullet \vec{U}\right)\vec{U} = - \nabla p + \left\{\left(\nu + \nu_t\right)\left(\nabla \bullet \vec{U} + \nabla \bullet \vec{U}_t\right)\right\} \tag{2}$$

Where ***U*** is the fluid velocity vector, ***p*** is the density-normalized pressure, ***ν*** is the kinematic viscosity of the fluid, and ν_t is the turbulent viscosity of the fluid. OpenFOAM provides three different pressure–velocity coupling methods for solving these equations: PISO (pressure implicit with split operator) (Issa, 1986); SIMPLE (semi-implicit method for pressure linked equations) (Patankar and Spalding, 1972); and PIMPLE, which is a hybrid of PISO and SIMPLE. The SIMPLEC (SIMPLE consistent) algorithm is also available, but only as part of the pressure based compressible solver rhoSimplecFoam. Since much detailed information is available in (Jasak, 1996), the following provides only a summary of the salient aspects of the incompressible numerical methods and models. Boussinesq (1877) hypothesized that the Reynolds tensor is a linear function of the mean velocity gradient, for incompressible flows, with eddy viscosity μt being analogous to the dynamic viscosity in the shear stress equation. Equation (1) defines the eddy viscosity as the product between density, a characteristic length scale of turbulence (l) and velocity V_t.

$$\mu_t = \rho \cdot V_t \cdot l \tag{3}$$

Where V_t, in most models is given by the root of turbulent kinetic energy (k) (Moukalled et al., 2016).

A comprehensive list of the numerical methods available in OpenFOAM can be found in (Robertson et al., 2015). The following provides some details of selected discretization methods used in the study.

2.5 Incompressible PISO/PIMPLE/SIMPLE solvers

The governing equations are generally solved using standard pressure–velocity coupling methodology- (1) momentum predictor, (2) pressure solver, (3) momentum corrector. The PIMPLE algorithms a unique variation of the PISO method, where an outer correction loops, i.e., cycling over a given time step for a number of iterations, and equation underrelaxation between outer correctors are allowed for stability. If no outer corrector loops are used, the algorithm is directly equivalent to the PISO method. PIMPLE solver also includes dynamic time-stepping (automatic time step adjustment to maintain a certain CFL number). The simpleFoam solver is based on the SIMPLE algorithm. It pursues a steady-state solution with the aid of under-relaxation factors between iterations. Equation under-relaxation helps promote diagonal dominance by boosting the influence of the owner cell terms (Jasak, 1996). Given the need to incorporate stratification into the model, we use a quasi-compressible approximation, including density as an explicit variable in the calculation. A detailed view of the algorithms used in OpenFOAM to couple pressure and velocity in the compressible case (PISO type algorithms, Pressure-Implicit with Splitting of Operators), can be found in (Paulo J. Oliveira, 2001). The great advantage of OpenFOAM is that each one of the above terms can easily be incorporated or eliminated from the solver that is used for each simulation, given that each application can be compiled independently. In summary, the solver includes the following equations: continuity, momentum conservation (NavierStokes), enthalpy conservation, ideal gas state and passive scalar transport, using the libraries and tools provided by OpenFOAM to solve them (thermophysical, turbulence and finite volume libraries). For Arya and Shipman (1981) work this study uses the model based on the SIMPLE algorithm, which in OpenFOAM is simpleFOAM.

2.6 Geometry and meshing

Given our interest in working with complex geometries, we use the GMSH tool, although OpenFOAM offers the snappyHexMesh alternative OpenCFD (2010) to solve the mesh generation problem. GMSH has proven to be very versatile when applied to different domain configurations. In particular, it allows STL files to be used as precursors for mesh generation, enabling the inclusion of complex geometric shapes and even topography in simulations. In most cases we use GMSH itself which is free software, despite the flexibility of working with files from other software such as STL format. The meshes consisted of hexahedra (hex) and split-hexahedra (split-hex) cells. In cases with complex geometry the mesh was refined near the walls to produce cells with wall normal dimensions between $y1+=10$ and $y1+=300$ adjacent to the surface. However, it is impossible to satisfy these criteria everywhere when processes of flow separation and attachment occur.

2.7 Geometry and meshing

Given our interest in working with complex geometries, we used the snappyHexMesh tool to solve the problem of mesh generation. SnappyHexMesh proved to be very versatile when applied to different domain configurations [40]. In particular, it allows STL files to be used as precursors to mesh generation, making it possible to include complex geometrical forms and even topography in the simulations. In the majority of cases we used the free CAE software Salome to generate the necessary STL files. The meshes consisted of hexahedra (hex) and split-hexahedra (split-hex) cells. In cases with complex geometry the mesh was refined near the walls to produce cells with wall normal dimensions between $y_{,}^{+} = 10$ and $y_{,}^{+} = 300$ adjacent to the surface. However, it is impossible to satisfy this criteria everywhere when processes of flow separation and attachment occur.

2.8 Treatment of walls

OpenFOAM offers a series of specialized libraries to define the boundary conditions on a surface. It has been shown Tominaga et al. (2008) that in the case of complex configurations that include a set of buildings or obstacles, it is preferable to use a smooth wall condition, given that a rough wall model requires a large grid, which makes it difficult to capture the details of the flow in complex geometries. Although different works have shown rough walls to be crucial in maintaining the correct ABL profiles along the upwind fetch, in this work any effect of the surface roughness is assumed to be small compared with the effects of the terrain, as suggested in (Silvester et al., 2009). There are many considerations to take into account when using wall functions in RANS simulations, considering the advantages and limitations of each wall model. For an extensive review, see (Blocken et al., 2007). However, the results shown in this work correspond to Low Re wall function using URANS simulations to compare results.

2.9 Initial and boundary conditions

In CFD the initial and boundary conditions used are of fundamental importance, because they directly affect the evolution of the simulation and must necessarily take into account the physical processes that we are modelling. Although OpenFOAM offers a wide variety of pre-defined conditions, in the majority of case, it is necessary to create specific conditions for each simulation. Furthermore, there exist important differences whether the simulation considers buoyant effects or not. There are different alternatives to define the inlet velocity profile, depending on the information available. If we have experimental data, we can use the expression:

$$U(z) = \frac{u_*}{\kappa} ln\left(\frac{z+z_0}{z_0}\right) \quad (4)$$

Where $\boldsymbol{u^*}$ (friction velocity) and $\boldsymbol{z_0}$ (roughness height) are known or can be estimated from data and κ is a constant with approximate value of 0.4. Richards and Hoxey (1993) proposed inlet boundary conditions consistent the k-ε model defined by Eq. 5 for streamwise velocity and Equations 6 and 7 for turbulent kinetic energy and dissipation rate of turbulent energy, respectively:

$$k = \frac{u_*^2}{\sqrt{C_\mu}} \quad (5)$$

$$\varepsilon = \frac{u_*^3}{k\left(z+z_0\right)} \quad (6)$$

Where ***Cμ*** is a model constant. In URANS simulations this profile is maintained fixed as a forcing during the entire simulation. The outlet profile of velocity is defined as zero gradient, assuming a fully developed flow. The top boundary condition is normally set as zero gradient due to its practicality, however it is inconsistent with inlet Richard and Hoxey conditions, except under certain circumstances (O'Sullivan et al., 2011). A constant shear stress should be applied as the. domain is sufficiently small to be entirely inside the constant shear stress layer. As reported in O'Sullivan et al. (2011) work this condition return the von Neumann conditions as the following equations:

$$\frac{\partial U}{\partial z} = \frac{u_*}{k\left(z+z_0\right)} \quad (7)$$

$$\frac{\partial \varepsilon}{\partial z} = \frac{u_*^3}{k\left(z+z_0\right)^2} \quad (6)$$

Already the lateral contour condition was used an empty condition, which is suggested by (OpenCFD, 2016) for two-dimensional simulations.

2.10 **Turbulence models**

The turbulence reference models made in this study are separated into two groups. The first is based on turbulent viscosity models, which contains the standard k-ε, k-ε realizable and RNG k-ε models. In addition to these, there are two more models based on the specific turbulence dissipation rate ω, these are the k-ω model and another model implemented in OpenFOAM as a variant of the SST k-ω model by Menter et al. (2003). The second group used is based on Reynolds stress modeling, where only one model was used in this work, the SSG Reynolds Stresses. For the k-ε, RNG k-ε, k-ω and SST k-ω models, the results are also compared with those obtained numerically by Martins et al. (2003).

The standard k-ε model is one of the most prominent turbulence models and has therefore been implemented in CFD for various purposes as it is considered a standard model for industrial cases, proving to be stable and numerically well constituted. Furthermore, it has good accuracy. For general simulations, this model offers good performance in terms of approximation. However, there are some applications for which this model is not suitable, which may include: boundary layer separation flows; flows with sudden changes in bottleneck rates; flow in rotating fluids; flow on curved surfaces.

The RNG k-ε model is an alternative to the standard k-ε model, in general, it offers an improvement in the size of the boundary layer recirculation compared to the standard k-ε model. The Realizable k-ε, developed by (Shih et al., 1995), differs from the standard k-ε model in two ways. Firstly, it contains a new formulation for turbulent viscosity: Cμ is not a constant as in the standard model, but a variable. The second difference is a new transport equation for the dissipation rate, which is derived from an exact transport equation for the mean square fluctuation of vorticity.

As a result, it mainly provides improvements in capturing the mean flow of complex structures and for flows involving rotation, boundary layers under strong adverse pressure gradients, separation and recirculation. The k-ω model is an alternative for predicting flows with low Reynolds numbers, while the SST model has a mixing function that applies the k-ω model in regions close to the wall and in regions where there is recirculation far from the wall, the standard k -ε model.

Finally, the SSG Reynolds Stress model, proposed by Speziale et al. (1991), was chosen because it did not need to modify the meshes used in this study, making it possible to maintain the same configuration for all models. These models use specific Reynolds transport for their formulation. They are responsible for the direct effects of Reynolds stresses and complex interactions in turbulent flows. Reynolds stress models offer significantly better accuracy than turbulence models based on eddy specificities, while being computationally cheaper than DNS and LES methods.

3. **METODOLOGY**

Numerical tests were reproduced to verify the size of the recirculation and velocity profiles obtained experimentally by Arya and Shipman (1981), in addition to this, other turbulence models were tested. The experimental results of Arya and Shipman (1981) were obtained in a wind tunnel with a section of 1.8 m wide, 1.5 m high and 11 m long, where the intensity and size of the turbulence scales were controlled to a neutral atmospheric boundary layer. The uncertainty regarding the results obtained was 15%. The geometric parameters of the triangular surface have height H = 0.15m and base width B = 0.385m, as shown in Figure 1.

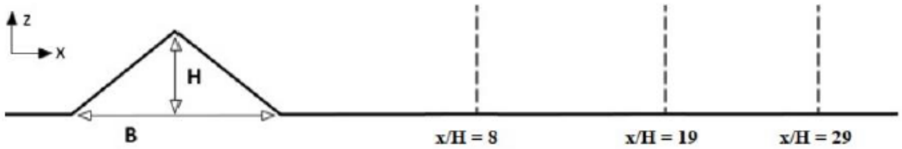

Figure 1. Out-of-scale graphical representation of the triangular surface with the locations of the experimental velocity profiles.

Arya and Shipman (1981) submitted a free stream speed of 8 m/s in a wind tunnel. The developed boundary layer reached a height of 1.5 m near the summit, therefore these reference values, together with the roughness height z0 of 0.25 mm, were imposed on the input via Eq. (5). The coordinate system has been changed to be a function of the height of the triangular surface (H). The origin, x = 0, was chosen to be the base of the triangle. As a result, a recirculation size of 13H was found and explored for the velocity path at positions x/H = 8, x/H = 19 and x/H = 29. To define the mesh, several mesh tests were performed with the standard k-ε model to define which mesh presented parameters close to those considered ideal by OpenCFD (2010) and eventual convergence with this model. The best mesh for this model was identified, which was applied to the other turbulence models, previously mentioned.

The models that did not converge with the mesh used in the standard k-ε model, due to differences in the distance from the first node to the wall, were treated separately, performing other mesh tests for these models. The solver used was simpleFoam based on the SIMPLE algorithm, in addition to using the boundary conditions provided for the atmospheric flow. The simulations were carried out on a Dell PowerEdge server and met the error criteria of 10−5 for the velocity field and 10−4 for the other quantities.

4. RESULTS

Table 1 shows the results obtained from the literature and the results of the turbulence models tested. The first line presents the experimental results obtained by Arya and Shipman (1981). In the second and third line are presented the numerical results obtained by Moreira et al. (2010) for the k-ε models and the RNG k-ε model. The subsequent lines of the table present the numerical results of the OpenFoam.

The recirculation size with the RNG *k-ε* and SST *k-ω* and SSG models is within the Arya and Shipman experimental error margin of 15%. Possible sources of error for the numerical results are the fact that there is an extreme condition of pointed obstacle that causes unorthogonality in cells near the summit (Moukalled et al., 2016; Mouzakis and Bergels, 1991), necessary for y+ configuration; the aspect ratio, that is, the difference between adjacent cell volumes near the summit and the fact that it was simulated in a 2D domain, which does not take into account a three-dimensional nature of turbulence.

Table 1. Recirculation size after triangular structure.

Table 1. Recirculation size after triangular structure.

Turbulence Model / Experimental Measurement	Recirculation Size
Experimental Arya and Shipman (1981)	13.0 H
k-ε Moreira et al. (2010)	8.7 H
RNG k-ε Moreira et al. (2010)	13.1 H
k-ω Moreira et al. (2010)	6.7 H
SST k-ω Moreira et al. (2010)	9.3 H
SSG[1] Moreira et al. (2010)	20.3 H
k-ε[1]	8.60 H
Modified k-ε[1]	8.40 H
Realizable k-ε[1]	11.47 H
RNG k-ε[1]	12.53 H
k-ω[1]	9.93 H
SST k-ω[1]	13.00 H
SSG[1]	13.01 H

(1) measured at 25°C

The results obtained for the models k-ε and RNG k-ε were inferior to those obtained numerically by Martins et al. (2003), but present differences smaller than 15%. The standard k-ε model presented results far from the experimental values of Arya and Shipman (1981), with differences of 33%, as also shown by Martins et al. (2003). Although it has a more robust implementation accounting Reynolds tensor anisotropy, the SSG model does not give very accurate results for near-wall velocity. Therefore, it has shown ability to predict recirculation size, as seen in Tab. 1, what was very more effective in this work rather than in Moreira et al. (2010) work, possibly due to either the applied top conditions or the used softwares.

For the profile at x/H = 8 the numerical results are very close to the experimental measurements from z = 1.8H. In experimental measurements below z = 0.5H, as pointed out by Mouzakis and Bergels (1991) and Kim and Patel (2000), there is an inconsistency with the fact that this profile "cuts" the recirculation zone, where it is expected negative values, being attributed to this inability the instruments used to detect direction. For the profiles at x/H = 19 and x/H = 29 (Fig. 2b and Fig. 3), the k-ω, k-ε and modified k-ε models performed in this work showed good agreement with the experimental results, where they have average errors less than 5%. However, they assumed the worst recirculation size values, with an error of 25% for the k-ω model. This fact is due to the ability of this model to act near walls and the inability to predict recirculation size. Even so, it presents an error about 50% smaller than the same model implemented with Ansys CFX in

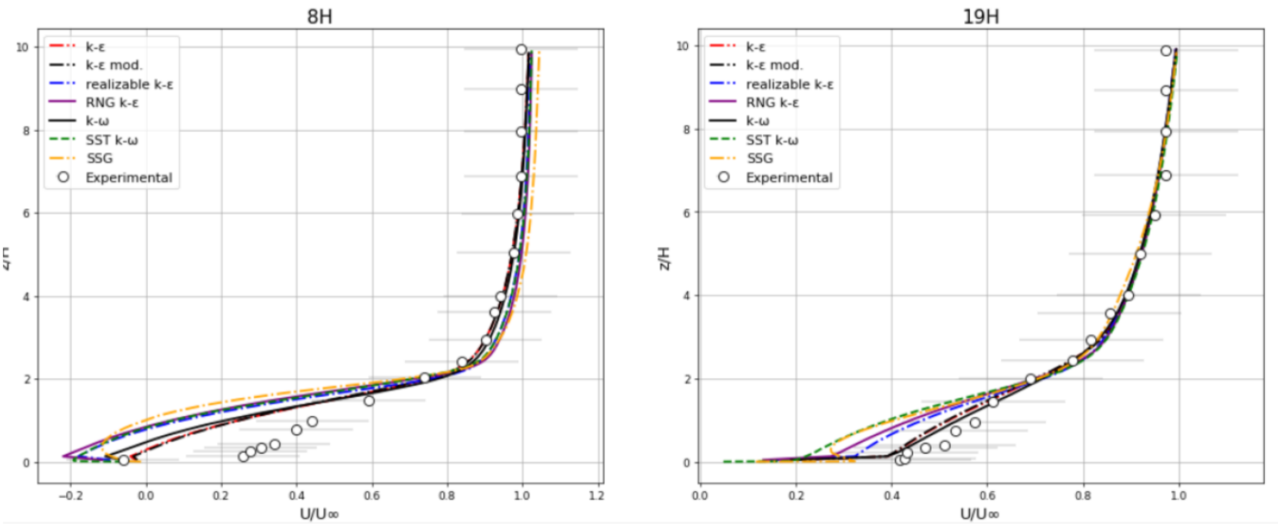

Figure 2. Velocity profile at (a) x/H = 8 and (b) x/H = 19.

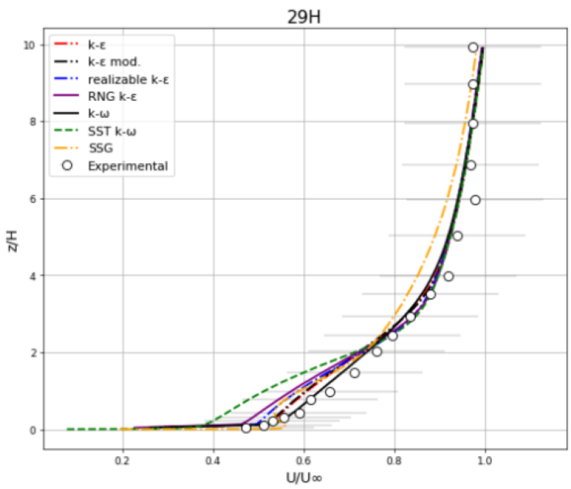

Figure 3. Velocity profile at x/H = 19.

5. CONCLUSIONS

The boundary conditions applied by Richard present in this work proved to be consistent when applied to the numerical model replicated in this work. Here, the method considered most appropriate is RNG k-ε, reinforcing this qualitative result also found by Kim and Patel (2000), denoting the full capacity of the OpenFOAM library in developing low-cost CFD analyses.

Reynolts stress models have good robustness, but require greater computational power and according to the literature, their application would become more viable if the model were 3D. The model is not relatively new and optimization of additional models could improve the model's prediction for non-neutral stratifications, but its relatively simple and free application could be a good option ahead of commercial solutions.

The modification of terrain roughness had minimal impact on flow behavior, possibly attributed to the analysis being conducted under steady-state flow conditions.

Future work involving complex geometries of real terrain will be studied with this software, which will be based on the choice of turbulence models present here, therefore, it could be useful for Investigating the Influence of Soil Roughness on Air Flow in the Atmospheric Boundary Layer.

6. Acknowledgements

The authors thank the UFPB for the resources and the maintenance of the scholarship to the student.

7. References

Arya, S. and Shipman, M., 1981. “An experimental investigation of flow and diffusion in the disturbed boundary layer over a ridge—i. mean flow and turbulence structure”. Atmospheric Environment (1967), Vol. 15, No. 7, pp. 1173 – 1184. ISSN 0004-6981. doi: https://doi.org/10.1016/0004-6981(81)90308-5. URL http://www.sciencedirect.com/science/article/pii/0004698181903085.

Abbasi, Ali; Annor, Frank Ohene; Van De Giesen, Nick. The effects of small water surfaces on turbulent flow in the atmospheric boundary layer: URANS approach implemented in OpenFOAM. Environmental Modelling & Software, v. 101, p. 268–288, 2018.

Ahmed Kabir, Ijaz Fazil Syed; Kanagalingam, Sivamoorthy; Safiyullah, Ferozkhan. Performance evaluation of air flow and thermal comfort in the room with Wind-Catcher using different CFD techniques under neutral Atmospheric Boundary Layer. Energy Procedia, v. 143, p. 199–203, 2017.

Azad, R.S., 1993. The atmospheric boundary layer for engineers, Vol. 17. Springer Science & Business Media.

Blocken, B., Stathopoulos, T. and Carmeliet, J., 2007. “Cfd simulation of the atmospheric boundary layer: wall function problems”. Atmospheric Environment, Vol. 41, No. 2, pp. 238 – 252. ISSN 1352-2310. doi: https://doi.org/10.1016/j.atmosenv.2006.08.019. URL http://www.sciencedirect.com/science/article/pii/S135223100600834X.

Barasa, Maulidi; Xuemin, Li; Zhang, Yi; et al. The impact of the atmospheric boundary layer on the asymmetric wake profile: A bivariate analysis. Sustainable Energy Technologies and Assessments, v. 53, p. 102563, 2022.

Boussinesq, J., 1877. “Essai sur la theorie des eaux courantes”. Mem. Acad. Sci. Inst. Fr., Vol. 23, No. 1, pp. 252–260.

Cao, Yong; Tao, Tao; Shi, Yujiang; et al. Large-eddy simulation of separated turbulent flows over a three-dimensional hill using WRF and OpenFOAM. Journal of Wind Engineering and Industrial Aerodynamics, v. 236, p. 105357, 2023.

Churchfield, M., 2010. “A description of the openfoam solver buoyantboussinesqpisofoam, national renewable energy laboratory, national wind technology cente”.

Flores, F., Garreaud, R. and Muñoz, R.C., 2014. “Openfoam applied to the cfd simulation of turbulent buoyant atmospheric flows and pollutant dispersion inside large open pit mines under intense insolation”. Computers & Fluids, Vol. 90, pp. 72–87.

Garcia, M. and Boulanger, P., 2006. “Low altitude wind simulation over mount saint helens using nasa srtm digital terrain model”. Third International Symposium on 3D Data Processing, Visualization, and Transmission (3DPVT’06), pp. 535–542.

Hargreaves, D. and Wright, N., 2007. “On the use of the k–e model in commercial cfd software to model the neutral atmospheric boundary layer”. Journal of Wind Engineering and Industrial Aerodynamics, Vol. 95, pp. 355–369. doi: 10.1016/j.jweia.2006.08.002.

Hussein, A.S. and El-Shishiny, H., 2009. “Influences of wind flow over heritage sites: A case study of the wind environment over the giza plateau in egypt”. Environ. Model. Softw., Vol. 24, No. 3, pp. 389–410. ISSN 1364-8152. doi: 10.1016/j.envsoft.2008.08.002. URL http://dx.doi.org/10.1016/j.envsoft.2008.08.002.

Issa, R., 1986. “Solution of the implicitly discretised fluid flow equations by operator-splitting”. Journal of Computational Physics, Vol. 62, No. 1, pp. 40 – 65. ISSN 0021-9991. doi:

https://doi.org/10.1016/0021-9991(86)90099-9. URL http://www.sciencedirect.com/science/article/pii/0021999186900999.

Jasak, H., 1996. Error analysis and estimation for finite volume method with applicationsto fluid flows. Ph.D. thesis, Imperial College, University of London.

Juretic, F. and Kozmar, H., 2013. “Computational modeling of the neutrally stratified atmospheric boundary ´ layer flow using the standard k– turbulence model”. Journal of Wind Engineering and Industrial Aerodynamics, Vol. 115, pp. 112 – 120. ISSN 0167-6105. doi: https://doi.org/10.1016/j.jweia.2013.01.011. URL http://www.sciencedirect.com/science/article/pii/S0167610513000378.

Kim, H.G. and Patel, V., 2000. “Test of turbulence models for wind flow over terrain with separation and recirculation”.

Boundary-Layer Meteorology, Vol. 94, pp. 5–21. doi: 10.1023/A:1002450414410.

Larsson, E.U., Cehlin, S.M., Openfoam, U. and Tapia, X.P., 2009. “Environment modelling of wind flow over complex terrain”.

Launder, B. and Sharma, B., 1974. “Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc”. Letters in Heat and Mass Transfer, Vol. 1, No. 2, pp. 131 – 137. ISSN 0094-4548. doi: https://doi.org/10.1016/0094-4548(74)90150-7. URL http://www.sciencedirect.com/science/article/pii/0094454874901507. Liou, T.H.S.W.W., Shabbir, A., Yang, Z. and Zhu, J., 1995. “A new k-ε eddy viscosity model for high reynolds number turbulent flows”. Computers and Fluids, Vol. 24, pp. 227–238.

Martins, M.A., Valle, R.M. and França, G.A.C., 2003. “Test of turbulence models for wind flow on the downwind slope of a 2d ridge model in neutral atmosphere.” 17th International Congress of Mechanical Engineering.

Menter, F., Kuntz, M. and Langtry, R., 2003. “Ten years of industrial experience with the sst turbulence model”. Heat and Mass Transfer, Vol. 4.

8. Responsibility notice

The author(s) is (are) the only responsible for the printed material included in this paper.