SEMARAK ILMU PUBLISHING

CFD Letters

Journal homepage:
https://semarakilmu.com.my/journals/index.php/CFD_Letters/index
ISSN: 2180-1363

CFD Analysis in the Mesh Modified Gas Diffusion Layer of a Proton Exchange Membrane Fuel Cell (PEMFC)

Blandy Berenice Pamplona Solis[1], Julio César Cruz Argüello[1,*], Isaias May Canche[1], Leopoldo Gómez Barba[2], Mayra Polett Gurrola[3]

[1] Department of Systems and Computation, TECNM/ Instituto Tecnológico de Chetumal, Av. Insurgentes 330, 77013, Chetumal, Q. Roo, México
[2] CUCEA, Universidad de Guadalajara, Periférico Norte N° 799, Núcleo Universitario Los Belenes, C.P. 45100 Zapopan, Jalisco, México
[3] CONACYT- TECNM/ Instituto Tecnológico de Chetumal, Av. Insurgentes 330, 77013, Chetumal, Q. Roo, México

ARTICLE INFO

Article history:
Received 23 March 2023
Received in revised form 20 April 2023
Accepted 21 May 2023
Available online 5 December 2023

Keywords:
Fuel Cell; CFD; GDL; Post-process; Python; ParaView

ABSTRACT

Computational Fluid Dynamics (CFD) software is well known for its application feasibility as well as reliable results in modeling electrochemical, thermal, and fluid transport processes. CFD has been used to investigate the phenomena involved in the operation of fuel cells, providing a large amount of data that must be analyzed to improve cell efficiency. This paper aims to demonstrate that programming can be used in the post-processing phase, using scripts in Python language to automate data analysis, based on the results of the simulation of oxygen transport in Polymer Electrolyte Membrane Fuel Cell (PEMFC). The OpenFOAM open-source CFD tool solved the fluid governing equations through the SIMPLE algorithm of three proposed Gas Diffusion Layer (GDL) case studies. In this work, an algorithm is presented to extract, compute and visualize the post-process results, supporting the GDL selection.

1. Introduction

A Polymer Electrolyte Membrane Fuel Cell (PEMFC) is an alternative to the issue of power generation. The main advantage of PEMFCs, during the energy conversion process, is that they do not release carbon dioxide when using hydrogen as fuel, unlike other sources based on fossil products [1–3]. However, the operating conditions of fuel cells must be regulated to achieve acceptable efficiency and avoid component degradation.

In situ data such as temperature, pressure, species flow and saturation are not easy to obtain during operation due to the highly reactive and confined environment of the cells. This causes a paucity of experimental data, which does not allow adequate understanding of these processes [4, 5].

Computer model simulation generates a large amount of data on the fluid transport behavior, providing support to reduce cycle times in design and to optimize fuel cells [6, 7].

* *Corresponding author.*
E-mail address: julio.ca@chetumal.tecnm.mx (Julio César Cruz Argüello)

https://doi.org/10.37934/cfdl.16.1.5567

CFD (Computational Fluid Dynamics) is widely used in research because it reduces experimental and manufacturing costs to improve fuel cell efficiency. In recent years, Commercial CFD as ANSYS FLUENT, COMSOL and STAR-CD have been used to simulate the PEMFC behavior [8–20], they include tools to solve numerical models through intuitive and configurable user interfaces. On the other hand, open-source CFD tools have remained in a second place because users require solid programming knowledge to modify the source code and to understand fluid dynamics and numerical solution methods to set correct values in the configuration files *et al.,* [21–23]. Despite these disadvantages, open-source CFD tools are an excellent option that provides advantages such as collaboration between members of the users' community, access to source codes, possibility of customizing the software, and decreasing acquisition costs in research projects. One of the most popular open-source CFD tool for the modeling of fuel cells is OpenFOAM, as evidenced by the works by Lozano *et al.,* [24], Novaresio *et al.,* [22], Beale *et al.,* [25], Imbrioscia *et al.,* [26], Wang *et al.,* [27], Kim *et al.,* [28], and Kone *et al.,* [29].

The need for more realistic computational models of the phenomena in fuel cells allowed the development of simulations fitted to the three-dimensional geometry of PEMFCs [30]. Dutta *et al*., [31] proposed a 3D numerical model using the finite volume method to solve the Navier Stokes equations for the flow channel in fuel cells. The model was implemented in the commercial FLUENT software modifying the available code through user-defined functions. Herlambang *et al.,* [32] conducted a numerical simulation of single PEMFC using COMSOL 5.1 under different conditions, obtained voltage-current density and power current density, as indicators of cell efficiency, and determined that the higher the humidity in the cathode layer, the limitation of mass transportation decreased.

Likewise, three dimensional (3-D) PEMFC models implemented with the open source CFD tool OpenFOAM were performed by authors like Valiño *et al*., [24] who studied the influence of the relative entry positions of hydrogen and oxygen on the distribution of gases. Several 3D configurations, under simplified working conditions, were considered. As their main result, they showed that the flow of the stoichiometric default gas (usually hydrogen) forces the pattern of the other reactive gas (oxygen). Imbrioscia and Fasoli [26] proposed different flow field designs of bipolar plates to predict flow behavior when changing width, depth, and shape of the distributing channels, they found that a greater volume impacted positively on flow. Kim *et al*., [28] performed VOF simulations to study the effects of channel cross section (rectangular and trapezoidal) on water removal in a straight gas channel. Kone *et al.,* [29] developed an open-source toolbox for predicting the distribution of physical quantities in a complete proton exchange membrane fuel cell using OpenFOAM. They adapted the model of a solid oxide fuel cell (SOFC), presented by Beale [25], to a PEM fuel cell. Although these papers highlighted the importance of CFD to modeling and simulation of fuel cells, they did not address the potential of programming in the post-process phase to exploit the data generated by the solver.

CFDs manage large amounts of data about velocity and pressure distribution due to the number of elements generated during the geometry meshing process, thus, the simulation process requires computational power [33]. In addition, the solution results need to be analyzed through post-processing to present them appropriately to the end user [34–37].

At present, robust tools have been developed for data visualization, which display data in multiple forms such as charts, tables, graphs, filters, and comparisons. The integration of visualization software in CFDs has allowed an increase in their popularity due to the ease of generating graphs of the simulation results [6]. Also, post-processing requires a significant amount of time to exploit the data generated by the solution [7,38].

The main objective of this work is to automate the analysis of simulation results of oxygen flow in a fuel cell generated in OpenFOAM. Automation was achieved by creating python scripts for the ParaView visualization tool.

2. Methodology

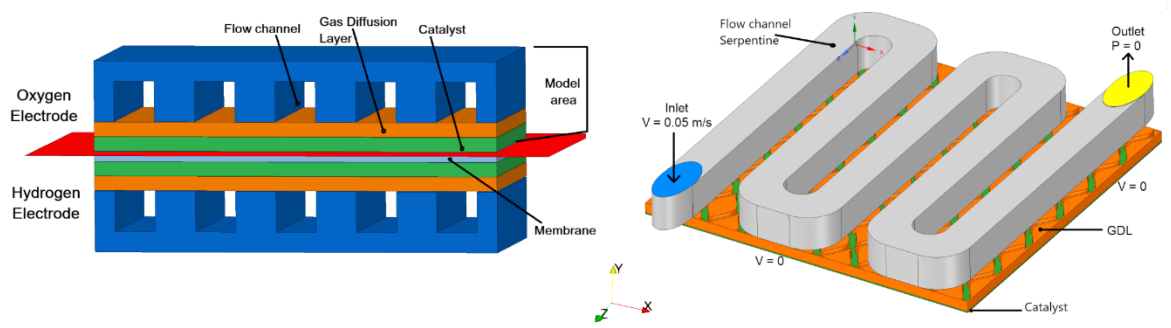
In general, CFDs establish three phases to solve fluid problems: a) Pre-processing, b) Solving, and c) Post-processing [34]. All these stages were adopted as part of the methodology used in the simulation process in this work, the procedure is described in the following section.

The focus of this research is the integration of programming techniques for the post-processing phase to enhance data analysis in the ParaView visualization tool. The generated results by the solver were accessed and processed using Python scripts, which managed the data to obtain images, graphics, comparisons, statistics that help understanding the behavior of oxygen flow in three different test cases of GDLs in fuel cells.

2.1 Pre-processing, Solving and Model Description

The analyzed data came from the simulation of oxygen transport through the cathode in fuel cells with Titanium GDL. In these simulations, three proposals were developed for the gas diffusion layer by modifying the contact geometry with the adjacent elements (serpentine, catalyst). One activity of the pre-processing phase is the definition of a computational domain and discretize it through the meshing process.

Figure 1 shows the geometric space used in fluid transport simulation through the oxygen electrode. It is composed of three main elements: 1) the serpentine, 2) the gas diffusion layer (composed of three Titanium meshes), and 3) the catalyst.



(a) Fuel cell model area (b) Computational domain

Fig. 1. Fuel cell components considered in the model

Superimposed Titanium meshes in different orientations were modeled to determine how the flow of oxygen presented a better behavior. Three test cases were made for the GDL: case A, three meshes placed at 0°-0°-0° positions (Figure 2a), case B, three meshes placed at 0°-45°-0° (Figure 2b), and case C, three meshes placed at 0°-90°-0° (Figure 2c). Table 1 presents the conditions used to develop the single-phase, isothermal, and steady state 3D model.

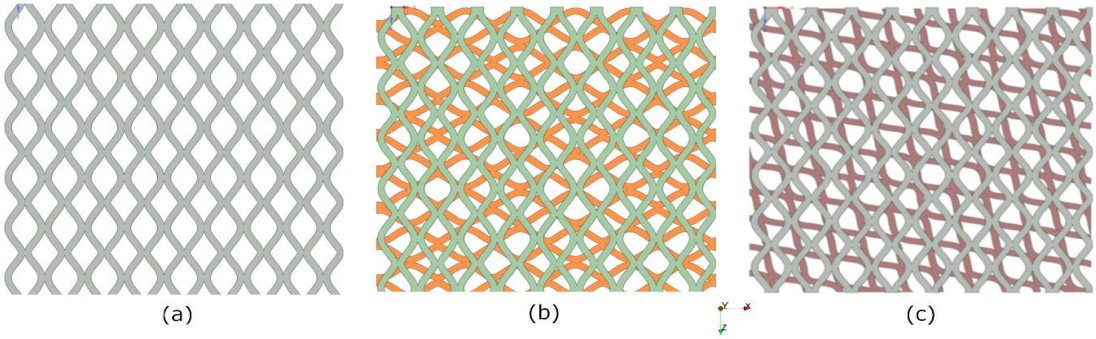


Fig. 2. Three test cases of GDL. (a) case A: 0°-0°-0°, (b) case B: 0°-45°-0°, (c) case C: 0°-90°-0°

Table 1
Conditions of the model context

Parameter	Value	Unit
Channel serpentine height	0.5	mm
Channel serpentine width	0.5	mm
Land area width	0.5	mm
GDL thickness	0.15	mm
Catalyst thickness	0.03	mm
GDL area	25	mm^2
Input velocity	0.05	m/s
Output relative pressure	0	Pa
Temperature	298.15	°K
Kind of fluid	Laminar	

The computational domain (Figure 2b) was segmented into a set of control volumes by the finite volume method to obtain a grid. In order to guarantee that the solution was independent from the grid size, a sensitivity analysis was conducted using a refinement technique. The fuel cell grid models were developed at different number of hexahedral and tetrahedral elements. Simulations for each case (A, B, C) were performed with three different domain sizes, shown in Table 2. In all cases, the same boundary conditions were established to calculate the velocity field. Numerical results (Table 2) showed that the maximum difference of case A for the solution of the velocity was less than 2%. In the same way, case B presented the greatest change for the average grid with a value of 0.68%. On the other hand, the smallest and medium grid belonging to case C, presented percentage differences of 0.87% and 1.08% respectively. These results are evidence of grid independence, since they presented differences of less than 3%, for which, they were considered acceptable [39].

Table 2
Results of the mesh sensitivity study

		Mesh 1	Mesh 2	Mesh 3	Maximum difference
Case A (GDL 0°-0°-0°)	Number of cells	837,564	1,159,792	2355,694	
	Velocity average	0.048918281	0.048836161	0.049787091	1.91%
Case B (GDL 0°-45°-0°)	Number of cells	1,355,716	1,684,409	3,206,470	
	Velocity average	0.045604228	0.045583106	0.04589628	0.68%
Case C (GDL 0°-90°-0°)	Number of cells	1,016954	1,655,296	2,545,741	
	Velocity average	0.043921236	0.043824086	0.044304686	1.08%

Once the pre-processing stage is carried out, it is necessary to run the solver for the solution of the equations governing the model. Each one of the three proposed cases (Figure 2b) in this article were solved using the simpleFoam solver integrated into the CFD OpenFOAM solution libraries. The SIMPLE (Semi-implicit Method for Pressure Linked Equations) algorithm is used by simpleFoam, a steady-state solver for incompressible or turbulent flow to solve the continuity equation Eq. (1) and momentum equation Eq. (2):

$$\nabla.\mathrm{u} = 0 \tag{1}$$

$$\nabla.(\mathrm{u} \otimes \mathrm{u}) - \nabla.\mathrm{R} = \nabla\mathrm{p} + \mathrm{Su} \tag{2}$$

where,

u = Velocity
p = Kinematic pressure
R = Stress tensor
Su = Momentum source

2.2 Post-process

OpenFOAM stores the obtained results in text files during the solution phase (solving stage). The post-processing tools present these simulation results through visualizations or summaries supporting end-user decisions during the new product design. So, the use of third-part applications as ParaView, EnSight and, Fieldview are required [40], which provide a graphical user interface to process the dataset, generating graphs, animations, and statistics. However, if the analysis of datasets requires multiple cases and large study zones, the post-processing can be a tiresome and time-consuming activity. The use of an Application Programming Interface (API) integrated into the visualization software improves post-processing. Languages such as C ++, Python, or Matlab can be utilized to program scripts and automate repetitive tasks.

The comparative study of the three test cases required an analysis of the average volumetric flow, velocity, kinematic pressure, and total pressure in multiple fuel cell regions throughout y-axis. Figure 3 shows the pseudocode to compute average values from surface slices, which are in-flow direction of the oxygen in the cell. The module libraries "Paraview.simple" were imported in Python scripts for the ParaView API to give functionality to the programming language and access the features of the visualization tool.

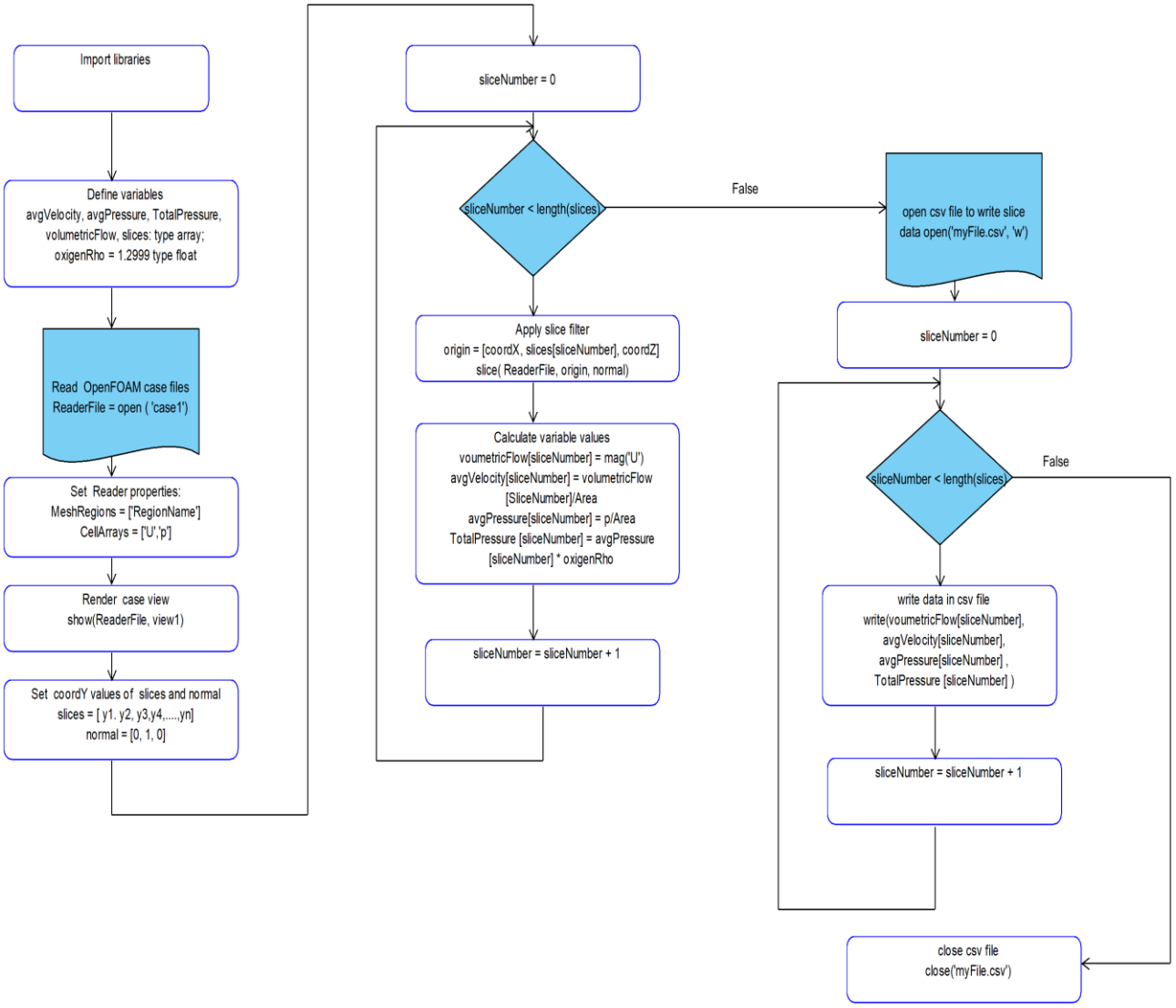


Fig. 3. Flowchart to develop the post-processing analysis

The algorithm starts importing the Paraview.simple library, defines the array variables for average velocity, average pressure, volumetric flow, total pressure, slice array, and oxygen density; then an openFoam file reader is created to access the data provided by the solver. After this, the 'segments' of the matrix are initialized with the values of the 'Y' coordinates to extract the slices from the cell surface and calculate the average values of volumetric flow, velocity, kinematic pressure and storage. of the total pressure in a "CSV" file for later use.

Figure 4 shows the graphical results of the script execution for case A. The transversal slices were generated every $5x10^{-5}$ mm across the cell. The developed Python script can be executed N-times and change the slice number parameter to analyze the interior of the fuel cell under different conditions. The digital repository urfcRepo/scriptingPython (github.com) contains the Python scripts and all experimental data for free access.

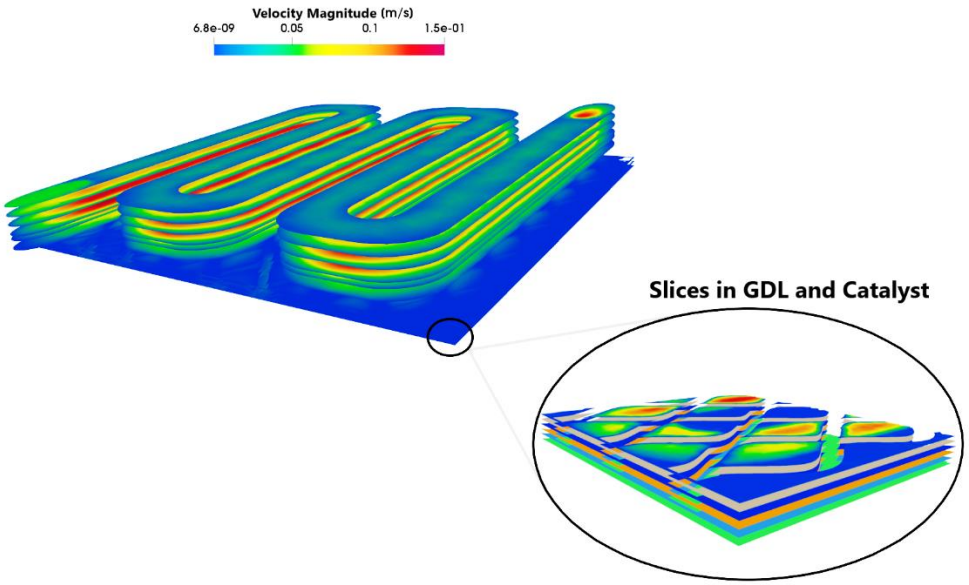


Fig. 4. Visual analysis of slices generation using Python scripting

3. Results

Velocity (U) and pressure (p) average values variables stored in Comma Separated Values (CSV) files were processed in Python using Pandas and Matplotlib libraries to access data structures and plot the results graphics of the average velocity behavior through the diffusion layer and the catalyst, as shown in Figure 5. Velocity decreases abruptly to position 0.5 mm due to the interface between the serpentine and the first mesh of the GDL. From this point, velocity has a gradual decrease until the second mesh edge at point 0.55 mm. So, in the cases B (0°-45°-0°) and C (0°-90°-0°), the velocity increases due to the change of the geometry produced by the mesh rotation. The last change is displayed at position 0.65 mm, where the GDL is in contact with the catalyzer.

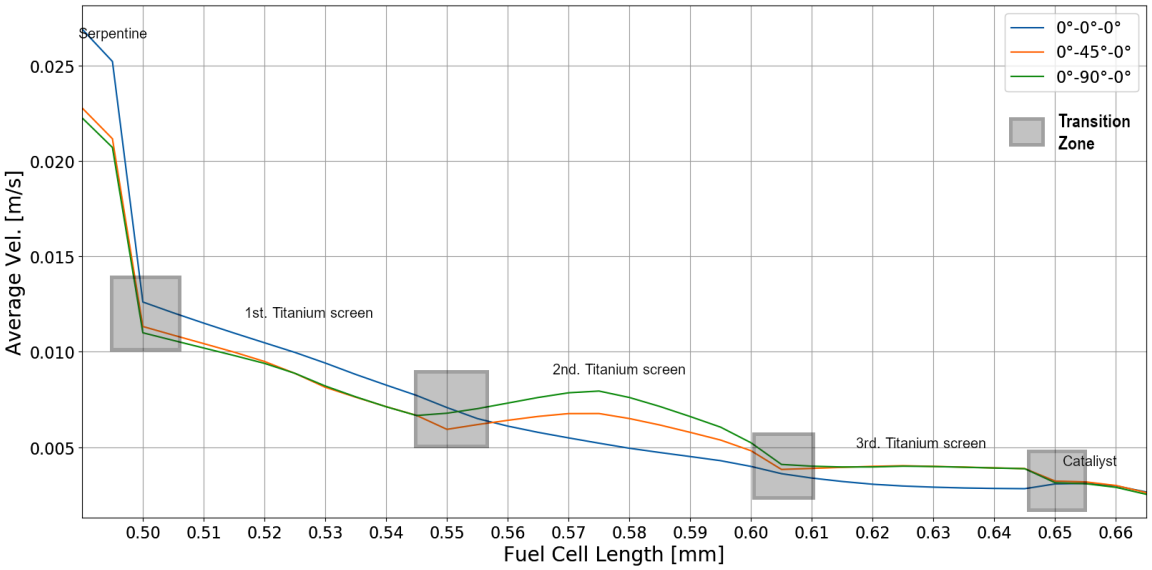


Fig. 5. In-flow Average velocity plot through fuel cell generated using Python script

In the same way, post-processing was applied using Python scripts to obtain stagnant zones where flow and pressure loss affect the fuel cell operation due to zero velocities. Figure 6 shows the stagnant areas for each proposed GDL, where values of less than 0.0001 m/s contribute to the flow stagnation. The gas diffusion layer of case B (0°-45°-0°) is the one that presented the lowest total stagnation area 0.0302630 mm^3 in contrast to the case A (0°-0°-0°) with a total of 0.0520527 mm^3 and case C (0°-90°-0°), which presented 0.0498623 mm^3. This may suggest that the distribution (0°-45°-0°) will present the highest efficiency in electricity production by improving the reactive phenomena inside the PEMFC.

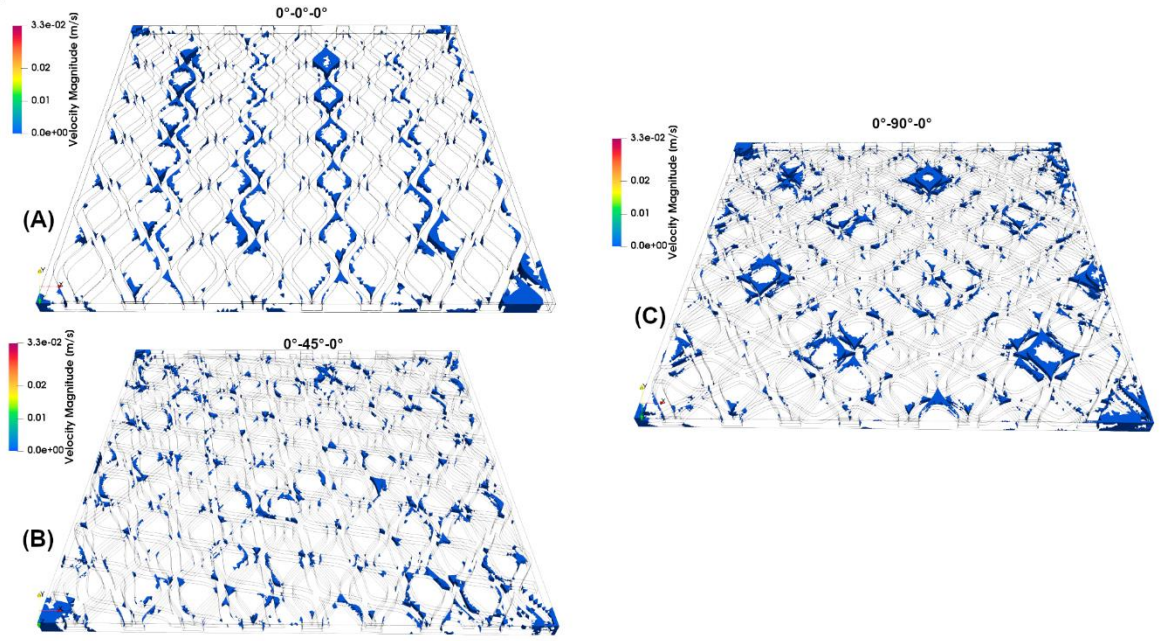


Fig. 6. GDL stagnant zones (marked with blue) found by means of Python script. A) GDL 0°-0°-0°. B) GDL 0°-45°-0°. C) GDL 0°-90°-0°

The average velocity magnitude was calculated considering the velocities of cases A, B, C to extract areas where the values are greater or equal than the mean and then compute the selected volume. These provide a metric to compare the flow of oxygen during its route to the catalyst. Figure 7 (left side) shows that the diffusion layer areas comply the condition "magnitude (velocity)> = 0.0061 m/s". Case B (0°-45°-0°) presented the largest coverage volume with a value of 0.7838283 mm^3 (see Table 2).

Also, the above process was applied to the catalyzer zone to filter the cells with “magnitude (velocity) >= 0.0028 m/s” condition. Figure 7 (right side) shows that case C presented the highest volume of 0.159214 mm^3 representing 32.04% of the catalyzer total volume (Table 3).

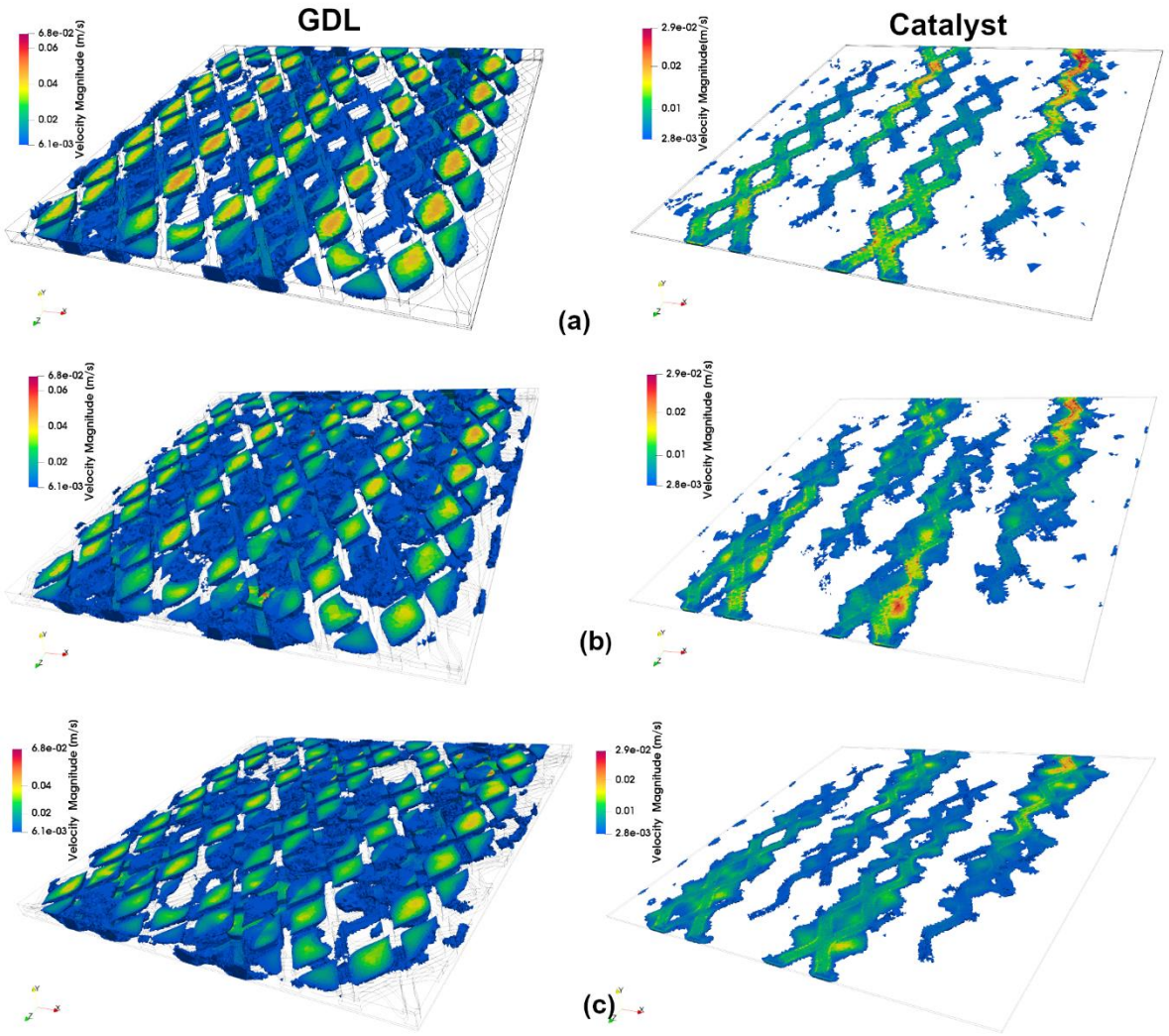


Fig. 7. Fuel Cell velocity zones greater than mean velocity. Left GDL, right Catalyst. (a) GDL 0°-0°-0°. (b) GDL 0°-45°-0°. (c) GDL 0°-90°-0°

Table 3
Volume zone velocities greater than average velocity (mm^3)

CASE	GDL total volume	Catalyst total volume	GDL cells volume>=0.0061	Catalyst cells volume >=0.0028
Case A. GDL 0°-0°-0°	2.31654	0.497	0.670008	0.123213
Case B. GDL 0°-45°-0°	2.29974	0.497	0.7838283	0.152865
Case C. GDL 0°-90°-0°	2.30904	0.497	0.749299	0.159214

The script executions provided information to analyze the behavior of the fuel cell, concluding that case B (0°- 45°- 0°) presented characteristics that improve performance due to lower velocity loss from the input, which benefits the chemical processes of the reactants. The average velocity in the gas diffusion layer for case B is 3.70% higher than the lowest velocity in case A.

The computing of the stagnant areas using Python scripts in the post-processing step provides fundamental support in cell fuel research at the pore level by enabling the highlighting of conflict

zones in cell operation. This research found that the GDL with lower volume was case B having 0.93% percentual points of less space than case A with a major stagnant zone.

Table 4 shows the criteria values to determine which one of the three diffusion layers presents operation advantages that impact fuel cell efficiency. Case B presents the best performance conditions: the lowest percentage of stagnation zones (1.32%), the highest mean velocity in the GDL (0.0034029 m/s), the percentage of zones volume greater than mean velocity in the GDL (34.08%) and the average velocity in the catalyst (0.00634029 m/s). The only criterion for which it is the second place is the percentage of zones with higher velocities in the catalyst (30.76%), being 1.28 percentage points below case C (32.04%).

Likewise, cases B and C presented better performances than case A, so it can be concluded that, concerning the velocity, the GDLs with rotation have better efficiency than those with meshes in only one direction.

Table 4
Evaluation criteria of the proposed GDLs

CASE	% GDL stagnant volume	% GDL volume greater zone velocities	% Catalyst volume greater zone velocities	GDL mean velocity m/s	Catalyst mean velocity m/s
Case A. GDL 0°-0°-0°	2.25%	28.92%	24.79%	0.006105	0.002855
Case B. GDL 0°-45°-0°	1.32%	34.08%	30.76%	0.006340	0.002888
Case C. GDL 0°-90°-0°	2.16%	32.45%	32.04%	0.006117	0.002744

4. Conclusions

The oxygen transport simulation through the cathode using CFD OpenFoam tool allowed to resolve fluid dynamic equations, generating a large amount of information that would have been difficult to obtain experimentally. The research aimed to demonstrate how to use Python script to facilitate the data's post-processing, helping to analyze and determine the best option from simulations carried out in a research process.

Post-processing tools use graphical interfaces to analyze data; however, when the processing requires more than one time, or multiple simulation cases, handwork becomes laborious, tiring, time-consuming, and there are high chances of making mistakes. The time to carry out a manual process can be reoriented to critical analysis of the visualization of results and calculated statistics through Python script to help support decisions on better products.

The use of Python scripting in the data exploitation is a key factor in saving the required time for the generation of results, graphics, and extraction of information from the simulations due to the provided flexibility and potentiality to the visualization tools allowing the process automation to obtain information on selected variables in the analysis of oxygen flow through GDL with different geometries, thus reducing the time required for the plotting results, selection, slices, and calculations.

In this work, the simulations are limited to the resolution of the velocity and pressure values, since the main purpose was to highlight the use of programming techniques to obtain insight for the future development of alternative optimization methods such as the non-dominated sorted genetic algorithm-II (NSGA-II) applied in the research work conducted by Xu *et al.,* [41].

The activities developed in this work are part of a wide field of research on the application of the fuel cells. Also, the possibilities to give continuity to the results obtained are highly feasible. In this

way, the use of information technologies supports the development of new simulations with different characteristics, as well as the implementation of programming knowledge in the post-processing phase, which is very useful.

Acknowledgments
We appreciate the support provided by the Doctorate in Information Technology of the Universidad de Guadalajara and the Tecnológico Nacional de México campus Instituto Tecnológico de Chetumal.

Conflicts of Interest
The authors declare no conflict of interest.

References
[1] Colleen Spiegel. *PEM Fuel Cell: Modeling and Simulation using MATLAB*. Elsevier; 2008. https://doi.org/10.1016/B978-012374259-9.50006-9.
[2] Shah, R K. 2007. "Introduction to Fuel Cells." In Recent Trends in Fuel Cell Science and Technology, edited by Suddhasatwa Basu, 1–9. New York, NY: Springer New York. https://doi.org/10.1007/978-0-387-68815-2_1.
[3] Barbir, Frano. 2013. *PEM Fuel Cells : Theory and Practice*. Second Ed. Sustainable World Series. Boston: Academic Press. https://doi.org/https://doi.org/10.1016/C2011-0-06706-6.
[4] Berning, T., D.M. Lu, and N. Djilali. 2002. "Three-Dimensional Computational Analysis of Transport Phenomena in a PEM Fuel Cell." *Journal of Power Sources* 106 (1–2): 284–94. https://doi.org/10.1016/S0378-7753(01)01057-6.
[5] Hossain, Mamdud, Sheikh Zahidul Islam, and Patricia Pollard. "Investigation of species transport in a gas diffusion layer of a polymer electrolyte membrane fuel cell through two-phase modelling." *Renewable energy* 51 (2013): 404-418. https://doi.org/10.1016/j.renene.2012.10.008.
[6] Tu, Jiyuan, Guan Heng Yeoh, and Chaoqun Liu. 2019. *Computational Fluid Dynamics*. London, UNITED KINGDOM: Elsevier. https://doi.org/10.1016/C2015-0-06135-4.
[7] Liu, Xiaofeng, and Jie Zhang. "Computational Fluid Dynamics: Applications in Water, Wastewater, and Stormwater Treatment: EWRI Computational Fluid Dynamics Task Committee." In *Computational Fluid Dynamics: Applications in Water, Wastewater and Stormwater Treatment*, pp. 1-224. American Society of Civil Engineers (ASCE), 2019.https://doi.org/https://doi.org/10.1061/9780784415313.
[8] Fontalvo, Victor, Danny Illera, Humberto Gómez, and Marco Sanjuan. "CFD Multiphysics Modeling and Performance Evaluation of PEM Fuel Cells." In *ASME International Mechanical Engineering Congress and Exposition*, vol. 58417, p. V006T08A067. American Society of Mechanical Engineers, 2017.https://doi.org/10.1115/IMECE2017-72160.
[9] Haghayegh, Marjan, Mohammad H. Eikani, and Soosan Rowshanzamir. "Modeling and simulation of a proton exchange membrane fuel cell using computational fluid dynamics." *International Journal of Hydrogen Energy* 42, no. 34 (2017): 21944-21954.https://doi.org/10.1016/j.ijhydene.2017.07.098.
[10] Ionescu, Viorel, and Nicolae Buzbuchi. 2017. "PEMFC Two-Dimensional FEM Model to Study the Effects of Gas Flow Channels Geometry on Reactant Species Transport." *Energy Procedia* 112 (October 2016): 390–97. https://doi.org/10.1016/j.egypro.2017.03.1085.
[11] Meng, Hua, and Chao Yang Wang. 2004. "Large-Scale Simulation of Polymer Electrolyte Fuel Cells by Parallel Computing." *Chemical Engineering Science* 59 (16): 3331–43. https://doi.org/10.1016/j.ces.2004.03.039.
[12] Yin, Yan, Xuefeng Wang, Xiang Shangguan, Junfeng Zhang, and Yanzhou Qin. 2018. "Numerical Investigation on the Characteristics of Mass Transport and Performance of PEMFC with Baffle Plates Installed in the Flow Channel." *International Journal of Hydrogen Energy* 43 (16): 8048–62. https://doi.org/10.1016/j.ijhydene.2018.03.037.
[13] Eldrid, Sacheverel, Mehrdad Shahnam, Michael T. Prinkey, and Zhirui Dong. 2003. "3D Modeling of Polymer Electrolyte Membrane Fuel Cells." In *Fuel Cell Science, Engineering and Technology*, 195–202. https://doi.org/10.1115/fuelcell2003-1719.
[14] Le, Anh Dinh, and Biao Zhou. 2009. "A Generalized Numerical Model for Liquid Water in a Proton Exchange Membrane Fuel Cell with Interdigitated Design." *Journal of Power Sources* 193 (2): 665–83. https://doi.org/10.1016/j.jpowsour.2009.04.011.
[15] Bednarek, Tomasz, and Georgios Tsotridis. 2017. "Issues Associated with Modelling of Proton Exchange Membrane Fuel Cell by Computational Fluid Dynamics." *Journal of Power Sources* 343: 550–63. https://doi.org/10.1016/j.jpowsour.2017.01.059.

[16] Iranzo, Alfredo, Miguel Muñoz, Felipe Rosa, and Javier Pino. 2010. "Numerical Model for the Performance Prediction of a PEM Fuel Cell. Model Results and Experimental Validation." *International Journal of Hydrogen Energy* 35 (20): 11533–50. https://doi.org/10.1016/j.ijhydene.2010.04.129.

[17] Kahveci, Elif Eker, and Imdat Taymaz. 2018. "Assessment of Single-Serpentine PEM Fuel Cell Model Developed by Computational Fluid Dynamics." *Fuel* 217 (April): 51–58. https://doi.org/10.1016/j.fuel.2017.12.073.

[18] Zhao, Jian, and Xianguo Li. 2019. "Oxygen Transport in Polymer Electrolyte Membrane Fuel Cells Based on Measured Electrode Pore Structure and Mass Transport Properties." *Energy Conversion and Management* 186 (April): 570–85. https://doi.org/10.1016/j.enconman.2019.02.042.

[19] Li, Shian, and Bengt Sundén. 2017. "Three-Dimensional Modeling and Investigation of High Temperature Proton Exchange Membrane Fuel Cells with Metal Foams as Flow Distributor." *International Journal of Hydrogen Energy* 42 (44): 27323–33. https://doi.org/10.1016/j.ijhydene.2017.09.014.

[20] Liu, Hui-Chung, Chien-Hsiung Lee, Yao-Hua Shiu, Ryey-Yi Lee, and Wei-Mon Yan. 2007. "Performance Simulation for an Anode-Supported SOFC Using Star-CD Code." *Journal of Power Sources* 167 (2): 406–12. https://doi.org/10.1016/j.jpowsour.2007.02.033.

[21] Vetter, Roman, and Jürgen O Schumacher. 2019. "Free Open Reference Implementation of a Two-Phase PEM Fuel Cell Model." *Computer Physics Communications* 234 (January): 223–34. https://doi.org/10.1016/j.cpc.2018.07.023.

[22] Novaresio, Valerio, María García-Camprubí, Salvador Izquierdo, Pietro Asinari, and Norberto Fueyo. 2012. "An Open-Source Library for the Numerical Modeling of Mass-Transfer in Solid Oxide Fuel Cells." *Computer Physics Communications* 183 (1): 125–46. https://doi.org/10.1016/j.cpc.2011.08.003.

[23] Kone, Jean-Paul, Xinyu Zhang, Yuying Yan, Guilin Hu, and Goodarz Ahmadi. 2017. "Three-Dimensional Multiphase Flow Computational Fluid Dynamics Models for Proton Exchange Membrane Fuel Cell: A Theoretical Development." *The Journal of Computational Multiphase Flows* 9 (1): 3–25. https://doi:10.1177/1757482X17692341.

[24] Lozano, A., L. Valiño, F. Barreras, and R. Mustata. 2008. "Fluid Dynamics Performance of Different Bipolar Plates. Part II. Flow through the Diffusion Layer." *Journal of Power Sources* 179 (2): 711–22. https://doi.org/10.1016/j.jpowsour.2007.12.095.

[25] Beale, Steven B., Hae Won Choi, Jon G. Pharoah, Helmut K. Roth, Hrvoje Jasak, and Dong Hyup Jeon. 2016. "Open-Source Computational Model of a Solid Oxide Fuel Cell." *Computer Physics Communications* 200: 15–26. https://doi.org/http://dx.doi.org/10.1016/j.cpc.2015.10.007.

[26] Imbrioscia, Gerardo Martín, and Héctor José Fasoli. 2014. "Simulation and Study of Proposed Modifications over Straight-Parallel Flow Field Design." *International Journal of Hydrogen Energy* 39 (16): 8861–67. https://doi.org/10.1016/j.ijhydene.2013.11.079.

[27] Wang, Jiatang, Jinliang Yuan, and Bengt Sundén. 2017. "Modeling of Inhomogeneous Compression Effects of Porous GDL on Transport Phenomena and Performance in PEM Fuel Cells." *International Journal of Energy Research* 41 (7): 985–1003. https://doi.org/10.1002/er.3687.

[28] Kim, Jin Hyun, and Woo Tae Kim. 2018. "Numerical Investigation of Gas-Liquid Two-Phase Flow inside PEMFC Gas Channels with Rectangular and Trapezoidal Cross Sections." *Energies* 11 (6): 1403. https://doi.org/10.3390/en11061403.

[29] Kone, Jean-Paul, Xinyu Zhang, Yuying Yan, and Stephen Adegbite. 2018. "An Open-Source Toolbox for PEM Fuel Cell Simulation." *Computation* 6 (2): 38. https://doi.org/10.3390/computation6020038.

[30] Sivertsen, B, and N Djilali. 2005. "CFD-Based Modelling of Proton Exchange Membrane Fuel Cells." *Journal of Power Sources* 141 (1): 65–78. https://doi.org/10.1016/j.jpowsour.2004.08.054.

[31] Dutta, S., S. Shimpalee, and J. W. Van Zee. 2000. "Three-Dimensional Numerical Simulation of Straight Channel PEM Fuel Cells." *Journal of Applied Electrochemistry* 30 (2): 135–46. https://doi.org/https://doi.org/10.1023/A:1003964201327.

[32] Herlambang, Yusuf Dewantoro, Fatahul Arifin, K. Kurnianingsih, Totok Prasetyo, and Anis Roihatin. 2021. "Numerical Analysis of Phenomena Transport of a Proton Exchange Membrane (PEM) Fuel Cell." *Journal of Advanced Research in Fluid Mechanics and Thermal Sciences* 80 (2): 127–35. https://doi.org/10.37934/arfmts.80.2.127135.

[33] Oro, Jesus Manuel F Fernandez. 2015. *Técnicas numéricas en ingeniería de fluidos: introducción a la dinámica de fluidos computacional (CFD) por el método de volúmenes finitos*. Edited by Primera Ed. Editorial Reverté. https://elibro.net/es/lc/udg/titulos/46736.

[34] Versteeg, H K, and W Malalasekera. 2007. *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Edited by Pearson Education Limited. Second Ed. Pearson Education Limited.

[35] Thévenin, Dominique, and Gábor Janiga. 2008. *Optimization and Computational Fluid Dynamics*. Edited by Dominique Thévenin and Gábor Janiga. *Fluid Dynamics*. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-72153-6.

[36] Anderson, J., E. Dick, G. Degrez, J. Degroote, R. Grundmann, and J. Vierendeels. 2009. *Computational Fluid Dynamics*. Edited by John F. Wendt. Third Ed. Vol. 28. Berlin, Heidelberg: Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-540-85056-4.

[37] Hassan, Y. 2017. "An Overview of Computational Fluid Dynamics and Nuclear Applications." In *Thermal-Hydraulics of Water Cooled Nuclear Reactors*, 1st Ed, 729–829. Elsevier. https://doi.org/10.1016/B978-0-08-100662-7.00012-9.

[38] Mukha, Timofey. 2018. "Turbulucid: A Python Package for Post-Processing of Fluid Flow Simulations." *Journal of Open Research Software* 6 (October): 1–5. https://doi.org/10.5334/jors.213.

[39] Bao, Zhiming, Zhiqiang Niu, and Kui Jiao. 2019. "Analysis of Single- and Two-Phase Flow Characteristics of 3-D Fine Mesh Flow Field of Proton Exchange Membrane Fuel Cells." *Journal of Power Sources* 438 (July): 226995. https://doi.org/10.1016/j.jpowsour.2019.226995.

[40] 2018. OpenFOAM. 2018. *The Open Source CFD Toolbox User Guide*. OpenCFD.

[41] Xu, Jiang-Hai, Han-Zhang Yan, Ben-Xi Zhang, Quan Ding, Kai-Qi Zhu, Yan-Ru Yang, Zhong-Min Wan, Duu-Jong Lee, Xiao-Dong Wang, and Zheng-Kai Tu. 2023. "Multi-Criteria Evaluation and Optimization of PEM Fuel Cell Degradation System." *Applied Thermal Engineering* 227: 120389. https://doi.org/10.1016/j.applthermaleng.2023.120389.