

# Prediction of a methane circular pool fire with fireFoam

Camilo Sedano<sup>1</sup>, Omar López<sup>1\*</sup>, Alexander Ladino<sup>2</sup>, and Felipe Muñoz<sup>3</sup>

<sup>1</sup>Department of Mechanical Engineering, Universidad de los Andes, 111711 Bogotá, Colombia

<sup>2</sup>Department of Mechanical Engineering, Universidad del Valle, Cali, Colombia

<sup>3</sup>Department of Chemical Engineering, Universidad de los Andes, 111711 Bogotá, Colombia

**Abstract.** In the present work, the *fireFoam* solver was used with Large Eddy Simulation (LES) and the Eddy Dissipation Concept (EDC) for modelling a medium-scale methane pool fire. A convergence analysis performed, showed that a 2 Million elements three-dimensional mesh, is good enough to attain good numerical results. By comparing the numerical results obtained, with the experimental ones, as well as numerical results from previous studies, it was proven that the *fireFoam* solver is able to obtain satisfactory results.

## 1 Introduction

Developments in fire safety engineering have focused on fire detection, suppression, structures' heating and smoke filling rates [1]. Classically, fire phenomena have been studied through experimental techniques, which can be highly expensive. Also, due to the random and destructive nature of fires, experimental results can be nearly impossible to replicate. Due to this, computational and numerical modelling of fire phenomena has received an increasing interest in recent years.

All the same, several difficulties arise when modelling fire dynamics. In particular, different phenomena are taking place simultaneously, such as: turbulent flow and turbulent mixture processes, thermodynamics, heat transfer (radiation) and chemical kinetics [2]. Furthermore, the fact that each of these phenomena considers different time and length scales, thus implying that there are several assumptions that must be applied in order to couple these. This is why, different models which simplify the combustion, chemical kinetics and fluid dynamics processes must be used [3].

In conflagrations (defined as an uncontrolled fire spread [2]), these simplifications lead to two main types of models: zone and field models. The first one separates the space in which the fire takes place into two zones: one where the products of the combustion process remain and another where the reactants remain, until consumed by the reaction. On the other hand, field models use Computational Fluid Dynamics (CFD) for reactive flows, in order to solve the coupled equations of Navier-Stokes with chemical kinetics (mixture fracture solution) [4].

Although field models have a higher mathematical complexity than zone models, the first are able to adjust to almost any geometric domain and constraints. Therefore, this field has recently drawn an increasing

interest, in order to develop reliable models which are accurate and exact enough to predict fires. Some known field models are the Flame Surface Density model developed by Trouvé [5], the Partially Stirred Reactor model of Chen [6] or the Laminar Flamelet Model initially proposed by Peters [7].

The Eddy Dissipation Concept (EDC) is one of the main combustion models used within field models. In it, Magnussen recommends a way to “relate the rate of combustion to the rate of dissipation of eddies” [8], by assuming that the rate of reaction depends on the mean concentration of a reacting specie, turbulent kinetic energy and its' dissipation rate [9]. The EDC began as a mode able to consider turbulent and momentum mixing, while taking into account the chemical kinetics. Most recently the interest in this model has shifted to develop an EDC which can use Large Eddy Simulation (LES) turbulence models instead of the Reynolds Averaged Navier-Stokes (RANS) models traditionally used [9-12].

In recent years there have been several studies regarding the simulation of pool fires. The experiment proposed by Tieszen in 2002 [13] has been one of the main cases studied. The simulations by Xin in 2008 [14] and Cheung in 2009 [15] using the Fire Dynamics Simulator (FDS) used Tieszen experiment as base case. Their results mainly show the flow's velocity fields with great details of the fluctuating velocity results. Another example is the study of Novozhilov and Koseki [16] who use the FIRE software to simulate the results obtained in different experimental analysis including the experiment by Weckman and Strong [17]. They report results for the evolution of temperature with time, as well as normalized flame heights and burning rates for pool fires of different fuels and sizes. Finally, in 2014 Chen [18] recreates computationally the results obtained by Weckman and Strong [17], in an attempt to test the capabilities of the LES models developed in the solver *fireFoam*. Chen developed a new model for radiative

\* Corresponding author: [od.lopez20@uniandes.edu.co](mailto:od.lopez20@uniandes.edu.co)

emission, obtaining different results for the flame heights, heat generation, temperature and emissive contours. However, some of these results do not agree completely with the experimental ones.

Hence, the main objective of this work is to predict the dynamics of a medium-scale pool fire and study the influence of the computational domain's discretization especially in the vertical (flame propagation) direction, when using LES turbulence modeling with the EDC as the combustion model. As test case, the experiment proposed by Tieszen et al in 2002 [13] was selected and simulated.

## 2 Modelling

### 2.1. Experimental setup

The experiment developed by Tieszen et al. consisted of a 1 m methane pool fire in the FLAME facility [13], was used as base case for the present numerical study. The research focused on taking velocity measurements using Particle Image Velocimetry (PIV), thus showing contours for average velocities in the axial and cross-stream directions, velocity fluctuations and root-mean squares contours. The PIV technique was used in order to obtain detailed results which could be easily compared to numerical solutions using LES turbulent models [13]. This case has been well documented in literature, since different authors have used it as a case study with which they could validate the numerical results obtained using different solvers (FDS [15], SFLM [19], CSE Model [20]).

The experimental setup consists of a 1 m diameter burner located at the center of a 6.1m cubic facility, 2.45 m above the floor. The methane was injected at a speed of 0.097 m/s to assure a total heat release rate of 2.07 MW. Further details can be obtained from the original article of the experiment [13].

### 2.2 Computational setup

The simulations were carried out using OpenFOAM – 2.4.0 (OF). This version was compiled with the *fireFoam* solver developed between FMGlobal and CFD Direct. This way, the combustion and radiation models to be used with the solver, can be implemented with the turbulent models available in OF. Also, the model can then be solved using the finite volume methods available in these libraries. The *fireFoam* solver uses the PIMPLE algorithm in order to solve the Navier-Stokes equations for compressible flows. This algorithm allows combining both the SIMPLE and PISO algorithm, such that it can use different correction equations for the pressure and velocity [9].

### 2.3 Computational models

To run the simulations, two LES turbulence models were selected and compared. First, the Spectral Eddy Viscosity (SEV) [21] model was used in order to

perform a mesh convergence analysis. This way, a mesh could be selected such that it could get the most accurate results, without incurring in higher computational cost. This mesh would then be used with the Smagorinsky turbulence model such that a comparison between the two models could be fairly done by using the exact same mesh.

First, the SEV model proposed by Menon was implemented. This turbulence model “accounts for the total energy transfer” by assuming that energy and energy transfer are in phase [21]. Since this model consists of only one equation for the turbulent kinetic energy, less computational cost is required for the solution of the model. Although this assumption is big enough to deviate the results from the exact equations [21], previous computational studies [18] have proven good agreement between the numerical and experimental results.

On the other hand, the Smagorinsky model suggests that the viscous stresses are proportional to the flows' strain rate [22]. Its main advantage is the fact that it uses an algebraic solution while other models (such as SEV) use a differential solution. This implies that the Smagorinsky model should also incur in a low computational cost. This model has been one of the main turbulence models used in the simulation of conflagrations [15, 19].

The Eddy Dissipation Concept proposed by Magnussen was chosen to model the combustion. The EDC models the turbulent mix as a function of the turbulent kinetic energy and its dissipation, the latter which is equal to the reaction rate. This leads to an expression of the mass fraction in terms of the two turbulent properties [23], such that the model only solves one additional transport equation for the fuel's mass fraction [9].

Regarding the chemical kinetics solution, the *irreversibleInfiniteReaction* model was used which is available in the OF library. This model takes into account only one reaction (equation 1) to solve the chemistry of the case at hand, basically a one equation equilibrium model. By using a one reaction model, the chemical kinetics solution is greatly simplified since it limits the number of species (five in this case) thus reducing the amount of iterations and reaction rate equations the solver must solve, therefore reducing the global computational cost of the simulation.



The finite volume Discrete Ordinates Model (fvDOM) was also included in the simulation in order to model the radiation heat transfer. In the present work, the model was implemented considering the gaseous phase as a gray gas participating media, while the Radiative Transfer Equation (RTE) is solved through a “loose coupling” method. This means that the RTE is solved using the solution of the previous time step (lagged in time). Then, the divergence of the radiative heat field is used as a source term for the energy equation which is solved in the PIMPLE (PISO) algorithm [24].

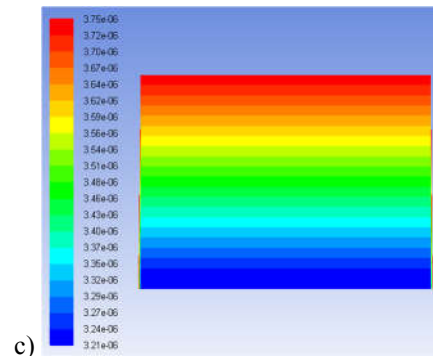
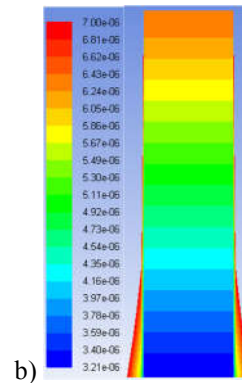
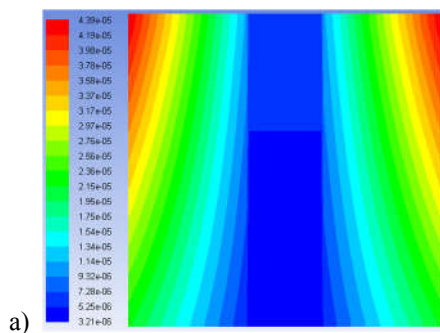
The simulations were set to have an adjustable time step dependent on the Courant-Friedrich-Lewy number (CFL). By using a CFL of 0.35, the simulation would adjust the time step such that the CFL fluctuates around 0.35. This way the entire model depends on the specified CFL rather than on a fixed time step, which will allow to solve the problem depending on the velocity of the fluid flow. Finally, the PISO algorithm will be solved using a Gauss Linear scheme, with an implicit Euler time discretization.

### 2.3 Computational domain, mesh and boundary conditions

The computational domain is a 3 m cube, according to the numerical simulation performed by Cheung et al [15], to make sure that the boundary conditions imposed by the facility are far enough from the flame, as well as disregarding the possible wall effects. The mesh was generated using OF's *blockMesh* utility, to create hexahedral elements. A total of 90 cells across the diameter of the pool fire were used [18], thus giving a total of 270 cells across the domain due to the restrictions from the geometry's vertices. As a first approach, four different meshes were generated in order to study the influence of the number of elements in the flame propagation (axial) direction. All of these were refined towards the bottom face, such that there are more elements near the burner's exit. Table 1 shows the main details of these meshes.

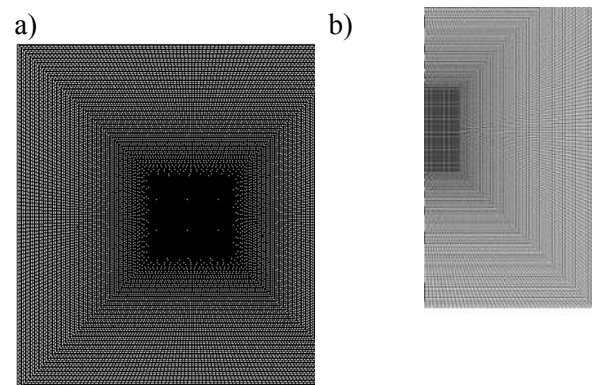
**Table 1.** Mesh descriptions.

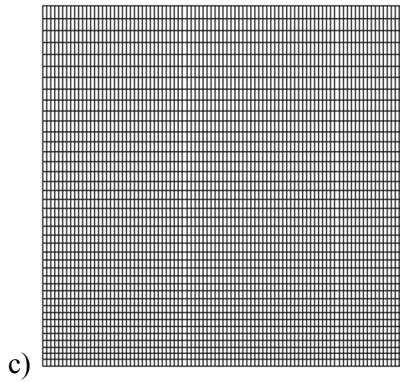
Mesh	No. Cells Base	No. of Cells Vertical	Total Cells [Millions]	Min. cell volume [m <sup>3</sup> x10 <sup>-7</sup> ]	Max. Aspect Ratio
Coarse (C)	270	40	1.62	32.14	28.91
Medium (M)	270	50	2.03	25.72	23.13
Fine (F)	270	60	2.43	21.43	19.28
Super-Fine (SF)	270	70	2.84	18.37	16.53



**Fig. 1.** Cell volume for mesh C. a) Midplane; b) Center region of midplane; c) Lower burner zone (30cm width x 30 cm height).

As shown by Figure 1 c), the smallest cell volumes are close to the burner's exit. This was intended in order to better capture the details of the combustion process of the reactive flow. Since the minimum cube-root-volume (CRV) for the coarse mesh is of 0.015 m, this could work as a first approach towards the size of the filter of the LES turbulence model to be used. Figure 2 shows the visualization and some details of Mesh C (1.62 million cells).





**Fig. 2.** Mesh a) Top view; b) Top view detail; c) Side view.

Four different boundaries were defined: the base, sides, outlet and an <inlet patch on the “base” face. The boundary conditions were set according to Table 2 and Table 3.

**Table 2.** Boundary conditions for pressure and velocity.

Boundary	Pressure	Velocity
Sides	calculated	<i>pressureInletOutletVelocity</i>
Outlet	calculated	<i>inletOutlet</i>
Inlet	calculated	0.097 m/s
Base	calculated	<i>zeroGradient</i>

**Table 3.** Boundary conditions for temperature, turbulent kinetic energy and mass fraction of methane and oxygen.

Boundary	T	k	CH <sub>4</sub>	O <sub>2</sub>
Sides	<i>inletOutlet</i>			
Outlet				
Inlet	275 K	1.41E-06 m <sup>2</sup> /s <sup>2</sup>	1	0
Base	<i>zeroGradient</i>			

The *zeroGradient* condition does not allow the variable to have any gradients in the normal direction to that face. The *inletOutlet* condition depends on the direction of the flow. When the flow leaves the domain, it behaves as a *zeroGradient* condition; however when the flow enters the domain (backflow), it performs as a constant value condition. The inlet values used for the *inletOutlet* boundary condition were:  $U = 0$  m/s,  $T = 300$  K,  $k = 1.41E-6$  m<sup>2</sup>/s<sup>2</sup>, CH<sub>4</sub> mass fraction = 0 and O<sub>2</sub> mass fraction = 0. The *pressureInletOutletVelocity* backflow, instead of having a constant value for backflow, it is calculated from the pressure obtained from the internal cell value [25]. The kinetic turbulent energy was calculated according to equation 2:

$$k = 1.5(UI)^2 \quad (2)$$

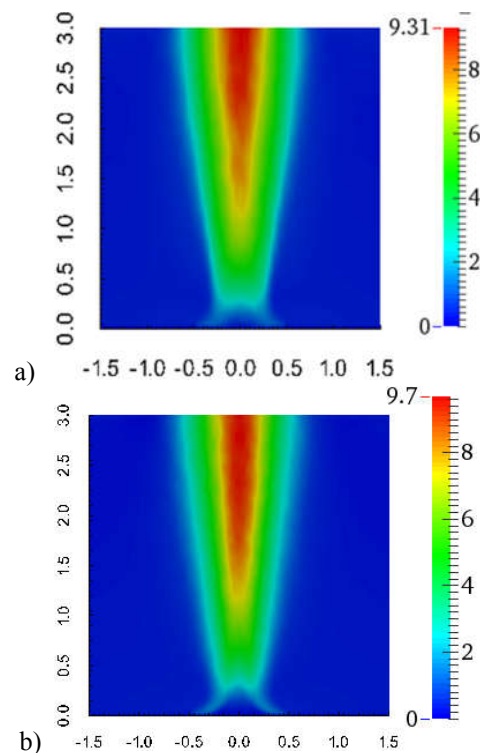
Where  $I$  is the turbulent intensity ( $I = 0.01$ ). At the “Outlet” boundary,  $k$  also has a value of  $1.41E-6$  m<sup>2</sup>/s<sup>2</sup>

when performing as an inlet condition. For the Smagorinsky turbulence model, instead of specifying a value for the turbulent kinetic energy  $k$ , the boundary condition must be specified for the kinematic viscosity  $\mu$ . In this case, a *zeroGradient* condition was used for this property at all boundaries.

### 3 Results and Discussion

#### 3.1 Spectral Eddy Viscosity model

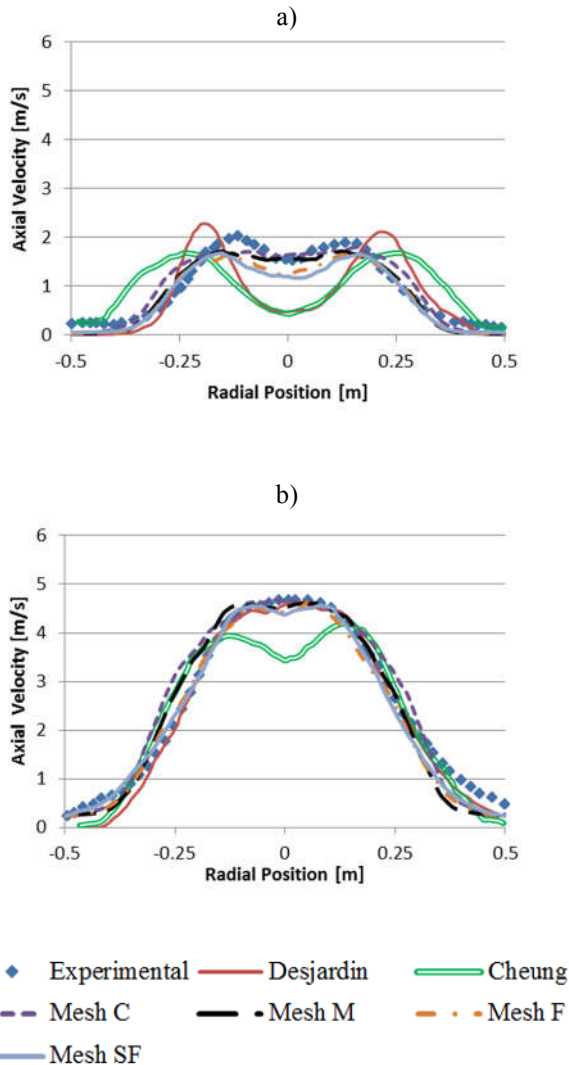
The results shown are based on time averages taken in a 15 s time frame (the last 15 s of the simulation).



**Fig. 3.** Velocity contours [m/s] for a) C mesh and b) SF mesh. (Domain scale in meters).

Figure 3 shows that SF mesh obtains average velocities up to 9.7m/s while the coarser mesh can only achieve values up to 9.3 m/s. Also, looking at the geometrical characteristics of the contours, the higher velocities (i.e. the red contours) begin at heights close to the 1.15 m mark for the C mesh, while for the SF mesh, these contours begin close to 1.3 m heights. The biggest difference however, appears at the bottom region of the domain. The 3 m/s contours are clearly more detailed in the finer mesh. This also appears to affect the cone-shaped geometry between the 0.5 m and 0.5 m widths up to the 0.3 m height. These differences between the contours of the different meshes are rather interesting, therefore data for axial and cross-stream velocity profiles at different heights were used to observe the convergent behavior of the simulations, as well as to compare them against the experimental results obtained by Tieszen et al

[13] and some numerical results from previous studies [15, 19].



**Fig. 4.** Axial velocity [m/s] vs radial position [m] at different heights; a) 0.2 m; b) 0.6 m.

From Figure 4 it can be observed that the general behavior of the simulations is very similar to the experimental results, especially for the 0.6 m height. Also, it is worth noting that the present implementation with fireFoam is able to obtain better results than the ones obtained by Cheung with FDS [15] or Desjardin with SFLM [19]. Finally, it is clear that there is a convergent behavior and particularly that Mesh M is able to obtain very accurate results without incurring in higher computational costs. To this regard, the maximum average axial velocity was selected as the main convergence criterion.

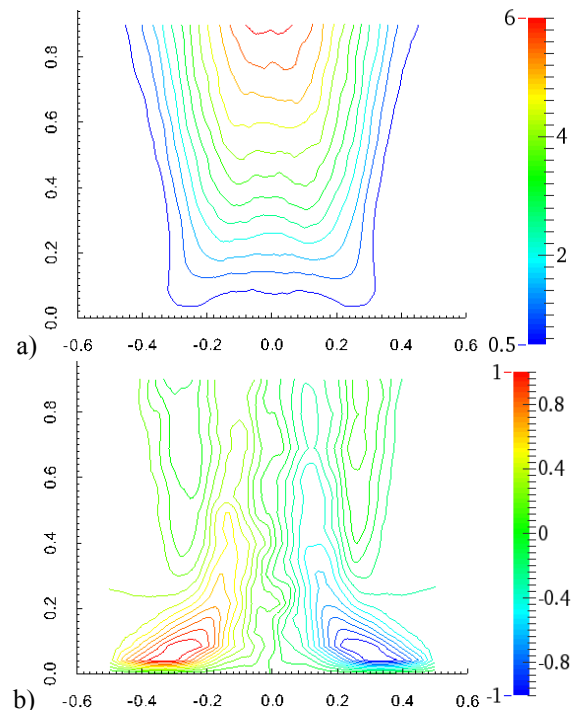
Table 4 shows the maximum average axial velocities for the four meshes at four different heights, as well as the average error for each mesh, when comparing to the experimental results. That is, the average of the error between the numerical result and the experimental result,

for each height. This shows that Mesh M is the one with the least average error, thus proving to be the most accurate one. Also, since it is the second coarsest mesh, it does show an advantage regarding computational cost. However, it also shows that all four meshes overestimate the top velocity at the 0.8 m height.

**Table 4.** Mesh convergence. Maximum axial velocity [m/s] at different heights.

Data	0.2 m	0.4 m	0.6 m	0.8 m	Error
Experimental	2.02	3.54	4.64	5.23	-
Mesh C	1.79	3.38	4.67	5.76	6.73%
Mesh M	1.71	3.47	4.61	5.67	6.63%
Mesh F	1.66	3.38	4.66	5.63	7.63%
Mesh SF	1.67	3.36	4.57	5.58	7.71%

Taking into account that the principal transport phenomenon is buoyancy, it is expected that the flow is dragged towards the centerline of the fire. This means, that there should be a symmetry towards the center plane (radial position 0 m), since the vertical transport drags the surrounding fluid creating the “necking” region at the bottom regions of the domain due to the entrainment phenomenon. Figure 4 does show this symmetric behavior expected, while Figure 3 shows the necking region at the bottom of the domain. This behavior can also be observed in Figure 5 which shows line contours for axial and cross-stream velocities for the bottom parts of the domain.



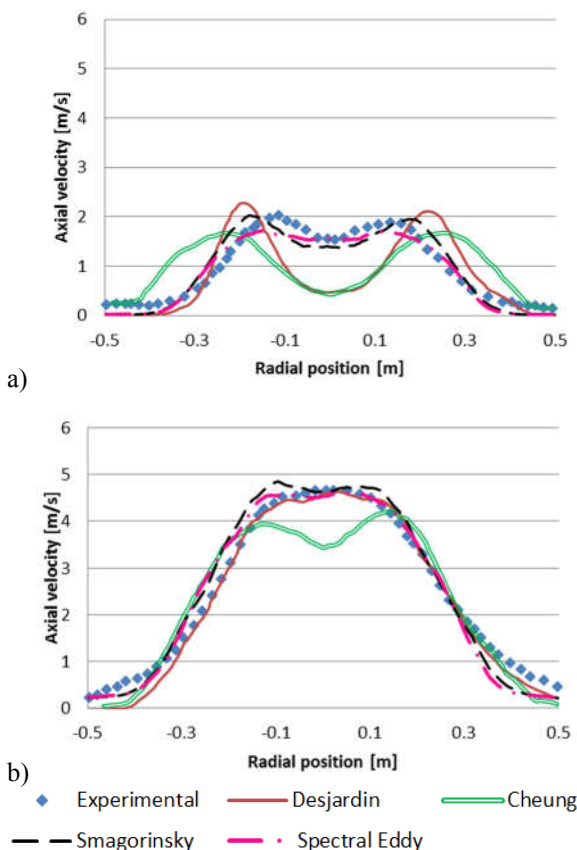
**Fig. 5.** Velocity line contours [m/s] for Mesh M; a) Axial velocity; b) Radial velocity (scales in meters).

Figure 5 a) shows the necking behavior expected from the poolfire, when looking at the curvature of the 0.5 m/s contour, which expands outwards from 0.3 m to

0.5 m widths. This matches the high velocity contours shown in Figure 5 b), which in turn show certain symmetry towards the centerline. Furthermore, both images help validate the use of the *fireFoam* software, since they show a very good resemblance with the experimental results obtained by Tieszen et al [13]. First, in Figure 5 a) the contours between 2.5 m/s and 3.5 m/s are close to the 0.4 m height. Second, the high velocity contours in Figure 5 b) (velocities above 0.8 m/s) go up to 0.2 m heights and between 0.2 m and 0.4 m wide. Similarly, the lower velocities (between -0.1 m/s and 0.1 m/s) are condensed in the central part of the domain, and towards the upper boundary.

### 3.2 Smagorinsky model

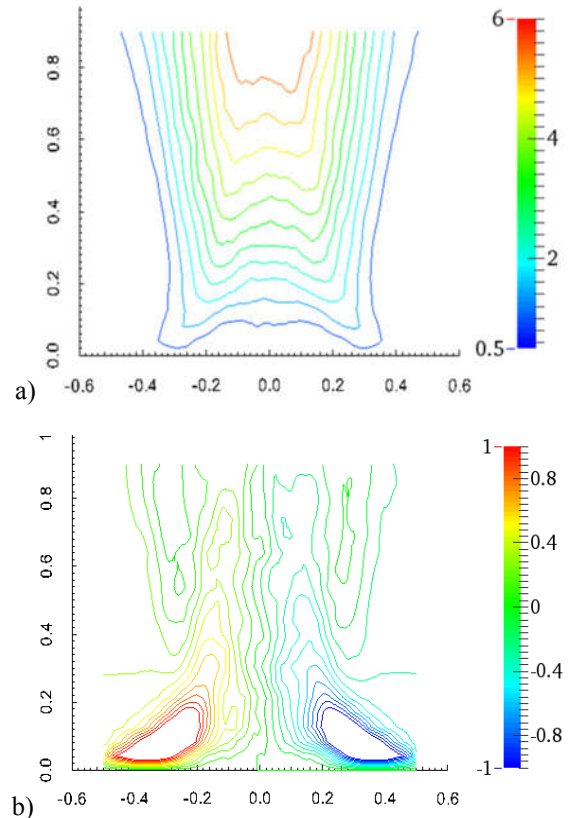
Previous research showed different studies [15, 19] which used the Smagorinsky LES model to account for the turbulent flow. Therefore, in the present study will compare the Spectral Eddy Viscosity proposed in the previous section and the Smagorinsky model. To do so, Mesh M (2.03M elements) was selected, taking into account that the results showed that this mesh was able to get accurate results without incurring in higher computational costs.



**Fig. 6.** Turbulent models comparison. a) Axial velocity at 0.2m height; b) Axial velocity at 0.6 m height.

Figure 6 shows the results obtained for the velocity in both the axial and cross-stream direction, against the radial position at different heights. From these images it is evident that the Smagorinsky model is able to obtain

better results than the Spectral Eddy Viscosity, when being compared to the experimental results. However, it is especially interesting to observe that the present implementation in *fireFoam* of the Smagorinsky model is able to obtain better results than other computational tools such as FDS [15] and SFLM [19].



**Fig. 7.** Smagorinsky – Velocity contours [m/s]; a) Axial velocity; b) Radial velocity (scales in meters).

Looking at Figure 7, and by comparing it to Figure 5, the Smagorinsky model does prove to obtain more detailed results. For example, the curvature of the 0.5 m/s axial velocity contour is more distinct towards the boundaries of the domain, as well as the curvature towards the centerline. This in turn, allows the other contours to have a peak close to the 0 m mark, which shows a better agreement with the experimental results. Also, looking at Figure 7 b) the higher velocity contours (red and blue contours) seem to be closer to each other, while maintaining the 0 m/s contour in the central part of the domain. This shows that the Smagorinsky model is able to better detail the phenomena at hand.

Finally, Table 5 shows that the simulation on the coarse (C) mesh took approximately 107 hours of CPU time running 16 parallel processes in an Intel Xeon processor E5-2695 v2, while the finer (SF) mesh took 285 hours of CPU with the same number of parallel processes. This again proves the necessity of the different models used, considering the complexity of the combustion process. Especially the turbulent and chemical kinetics models used are necessary to significantly reduce the computational cost. It is also interesting to notice that the implementation with the

Smagorinsky turbulent model (which uses the Medium mesh) also has a shorter computing time as the Spectral Eddy Viscosity implementation for the same mesh. This proves that not only the Smagorinsky model achieves better results, but also incurs in fewer computational costs.

**Table 5.** Computational cost.

Mesh	Processors	Time [h]
Coarse (C)	16	107
Medium (M)	16	298
Fine (F)	16	334
Super-Fine (SF)	16	385
Smagorinsky (M)	16	202

## 4 Conclusions

In the present work, simulations were carried out in order to replicate the experimental results obtained by Tieszen et al in a medium-scale methane pool fire, through the use of the open source code *fireFoam*. The numerical results show a good agreement with the experimental results particularly with axial (flame propagation direction) velocities and velocity line contours. Although, the numerical results for line contours are not geometrically exact to the experimental ones, the general behavior of these show that the models that were implemented can simulate effectively a pool fire. The regions with high velocities are similar in both cases (experimental and numerical), for both axial and cross-stream velocities. In particular, it is interesting to see the curvature of the axial line contours, which is a clear effect of buoyancy and entrainment, due to the high velocity areas shown in the cross-stream line contours. Also, mesh convergence was analyzed by running simulations with four different mesh sizes. Based on the convergence analysis, it is appropriate to affirm that a mesh with 90 divisions across the pool fire's diameter and 50 divisions in the axial direction (medium size mesh) is fine enough to obtain reliable results. By comparing the results of the 2.03M elements mesh using two different turbulent models, it was possible to confirm that the Smagorinsky LES model is able to achieve better results, while doing so in less computing time. Taking all of this into account, the *fireFoam* tool has proven to be an effective solver for the numerical simulations of medium scale pool fires. This can in turn lead to suggest the use of the computational tool for diffusion turbulent flames phenomenon of larger scales.

The authors would like to thank the Engineering schools of both, Universidad de los Andes and Universidad del Valle (Grant CI2868), for providing the resources and support for the development of the present research. Also, to the HPC support team at the Universidad de los Andes, for providing the cluster computing service in which the simulations took place.

## References

1. G. Hekestad, *Fire Safety J.*, pp. 25-32 (1984)
2. V. Novozhilov, *Prog. in Energy and Combustion Sci.*, **27**, pp. 611-666 (2001)
3. S. Turns, *An Introduction to Combustion*, McGraw-Hill, Singapore (2000)
4. B. Merci and T. Beji, *Fluid Mechanics Aspects of Fire and Smoke Dynamics in Enclosures*, CRC Press, Chennai, India (2016)
5. A. Trouvé and T. Poinso, *Proc. of the Summer Program 1992* (1992)
6. J.-Y. Chen, *Combustion Sci. and Tech.*, **122**, pp. 63-94 (1997)
7. N. Peters, *Prog. in Energy Combustion Sci.*, **10**, pp. 319-339 (1986)
8. B. Magnussen, and B.H. Hjertager, *Intl. Symposium on Combustion*, **16**, pp. 719-729 (1976)
9. H. K. Versteeg, and W. Malalasekera, *An introduction to Computational Fluid Dynamics. The Finite Volume Method*, Pearson Prentice Hall, Essex (2007)
10. S. Pope, *New J. of Physics*, **35**, pp. 1-24 (2004)
11. K. McGrattan, et al., *Fire Safety J.*, **30**, pp. 161-178 (1998)
12. Z. Chen et al., *Fire Safety J.*, **64**, pp. 12-26 (2014)
13. S. Tieszen et al., *Fire, Combustion and Flame*, **129**, pp. 378-391 (2002)
14. Y. Xin et al., *Combustion and Flame*, **153**, pp. 499-509 (2008)
15. S.C. Cheung and G. Yeoh, *Intl. J. of Thermal Sci.*, **48**, pp. 2187-2202 (2009)
16. V. Novozhilov and H. Koseki, *Combustion and Sci. Tech.*, **176**, pp. 1283-1307 (2010)
17. E.J. Weckman and A.B. Strong, *Combustion and Flame*, **105**, pp. 245-266 (1996)
18. Z. Chen et al., *Intl. J. of Heat and Mass Transfer*, **70**, pp. 389-408 (2014)
19. P.E. Desjardin, *Combustion Sci. and Tech.*, **177**, p.p. 1883-1916 (2005)
20. S. Ferraris et al., *Fire Safety Sci.*, **8**, pp. 963-974 (2005)
21. S. Menon et al., *Computers & Fluids*, **25**, pp. 165-180 (1996)
22. J. Smagorinsky, *Monthly Weather Review*, **91**, pp. 99-164 (1963)
23. B. Magnussen, *ECCOMAS Thematic Conference on Computational Combustion*, Lisbon (2005)
24. J. Tencer and J.R. Howell, *J. of the Brazilian Society of Mechanical Sciences and Engineering*, **119** (2015)
25. NEXTfoam, *Boundary Conditions – OpenFOAM-2.3.0* (2014)