



22nd Annual International Symposium
October 22-24, 2019 | College Station, Texas

Collision of Convex Objects for Calculation of Porous Mesh in Gas Explosion Simulation

Tatiele Dalfior Ferreira and Sávio Souza Venâncio Vianna*
University of Campinas
Albert Einstein 500. Campinas. SP – Brazil

*Presenter E-mail: svianna@unicamp.br

Abstract

We investigate the coupling of the flamelet combustion model with the collision distance algorithm for entertainment games. The collision algorithm is coded to calculate the porosity of the geometry based on the PDR (Porosity Distributed Resistance) approach for modelling of complex geometries. The turbulent field generated by the interaction of the flow with the porous objects is used to calculate the wrinkling length scale of the flame via the fluctuating velocities. The turbulent fluxes are amended in accordance with assigned porosities at the cell faces. The combustion and porosity models are implemented in the framework of an in house Fortran code that solves the full set of Navier-Stokes equations. Results are presented for non-reacting flows and reacting flows over a bluff body for $Re=44,000$ and $Ka=1$ (Reynolds and Karlovitz numbers, respectively). Numerical findings are compared with standard commercial CFD tools.

Keywords: Collision Algorithm, CFD, Vapor Cloud Explosion

1 Introduction

The numerical simulation of turbulent flows requires special attention to the so-called "small scales" objects. Since such objects increase the generation of turbulence, they must be considered in the flow modelling, which poses an extra challenge when building the computational grid. Even if Reynolds averaged Navier - Stokes equations are solved, the capturing of all details of the complex geometry leads to extremely expensive cost from the computational point of view [1]. When considering reacting flows, such deflagrations and detonations, the problem is even worst as the wrinkling of the area of flame front is highly dependant on the generation of turbulence. To overcome this limitation, researchers have pursued the parametrisation of the geometry based on PDR (Porosity Distributed Resistance) approaches [1] [2] [3].

Unfortunately, the representation of a complex geometry, such those commonly found in real engineering problems, is not straightforward. Little documentation is found where details on how the porous is calculated is presented. Most of the research works are focused on the results rather than providing details on how the porous representation of the geometry has been built.

We reasoned that the flux through the face of the computational cells as well as the rate of change in the control volume can be amended based on the collision of two objects as if the objects are treated as collision parts in video games that consider the collision between players [4] [5]. In order to develop the method that accounts for all details of the geometry, the geometrical model was built in a CAD tool ensuring that all parts of the geometry are treated as a set of triangles.

The stereolithography (STL) CAD files are widely used for quick prototyping as well as for computer-aided manufacturing. The main advantage is that the STL files describes the geometry as unstructured triangulated surface by the normal and nodes of the triangles using the three dimensional Cartesian coordinate system. On the other hand, the computational mesh was considered as a set of cubes. We bring forward an effective procedure to check the collision between the elements of the geometry (triangles) and the computational mesh (cubes) based on the Minkowski addition.

In this work, the stereolithography (STL) CAD files are treated as an individual convex set at the same time that the computational mesh is also treated as a second convex set. Although no geometry is considered in the computational mesh, every face of the computational cell and its respective volume were amended to take into account the presence of an obstacle at the same region of the Euclidian space as if it was placed at the same coordinate of the computational cell. As result of this methodology, a porous mesh is created where the Navier-Stokes equations can be numerically solved and the explosion simulation is performed.

2 Methodology

Considering two sets of vectors, say A and B, in the Euclidian space, the Minkowski sum can be defined by the sum of each vector in A to each vector in B leading to the following set:

The Minkowski addition concept was applied in a systematic manner to check whether there is collision between the elements of set A and elements of set B.

We first considered simple geometrical (Fig. 1a) model to examine if the details and contours of the geometry are well captured. We found that the porous (Fig. 1b) representation of the original geometrical model maintains the same dimensions and details of the faces and the edges that built up the topology of the geometry. The same is observed for the volume of the object. It is important to ensure that the faces of the computational cells are properly resolved as well as the volume of the computational since both entities will be used to calculate the flux of mass, moment and energy when solving the set of the equations governing the fluid flow.



Figure 1: The geometrical STL model (a) for a simple cube and its respective porous (b) model calculated using the GJK algorithm. Threshold of the porosity is presented at the region of the computational domain where the porosity is slightly lower than unity.

We thus explored more complex geometries which resembles real engineering devices. A typical chemical processing module (Fig. 2a) has been considered. Details of the decks and the three rounded equipment are well resolved by the porosity procedure. In contrast to the results presented so far, a detailed geometrical model (Fig. 3a) comprising a significant amount of equipment and details such small pipes and trails has also been considered. The original STL - CAD front view (Fig. 3a) can be compared with its twin model (Fig. 3b) where every single detail is captured by the novel technique. The perspective view (Fig. 3c) shows details of even smaller pieces of the geometry which are also fairly mimic by the approach as shown in Fig. 3b.

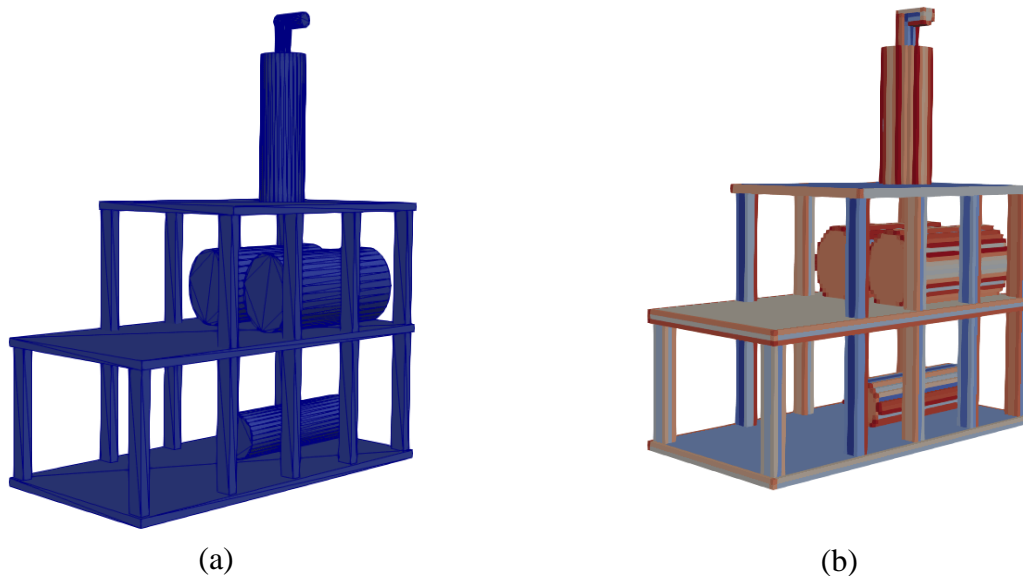


Figure 2: A chemical process module comprising two vessels and one heat exchanger at lower deck (a). Piping is omitted to ease the visualisation. The associated porous model (b) and details of the geometry well captured by the GJK algorithm applied to the STL model.

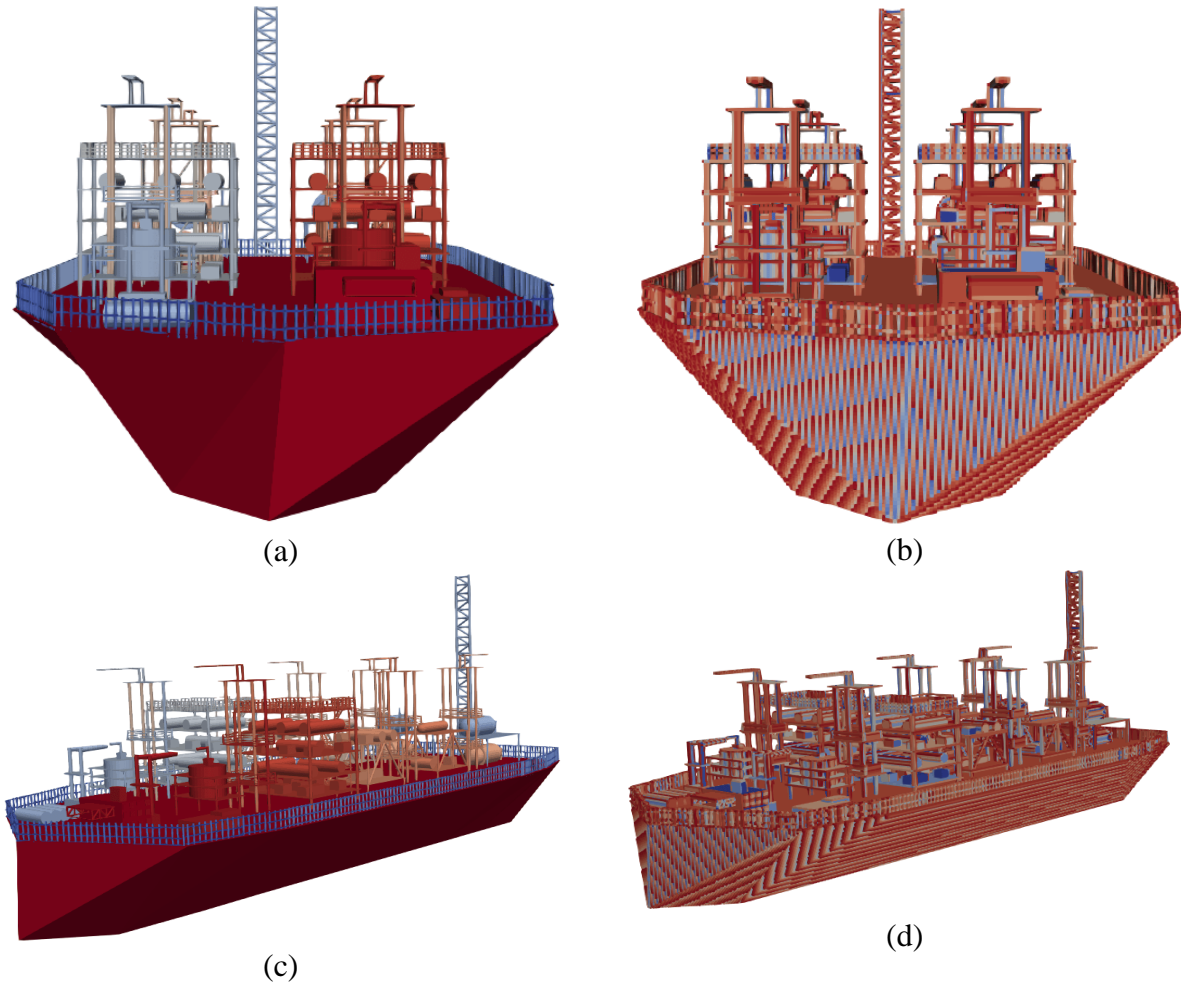


Figure 3: STL model from front view (a) and perspective view (c) of a complex geometry comprising several equipment as well as small details, such as pipings and valves. The associated parametrised porous model of the STL models for front view (b) and perspective view (d).

We further combined the porous regions of the Euclidean space, as represented by the porous model (Fig. 1b, Fig. 2b, Fig. 3b, Fig. 3d), with the finite volume method for solution of the set of Navier-Stokes equations. For the discretised computational cells, all cell faces and the volumes were amended in accordance with the obstructions caused by the porous model. We have followed the reasoning line that if the cell is completely blocked no flow is allowed. Should the cell be 50% blocked it would therefore allow half of the flow expected for a full open cell.

In doing so, porosity correction factors range from zero to unity. A blocked computational cell would be assigned zero porosity while a fully opened cell would be assigned unity porosity. The porosity values are used in the conservation equations to correct the fluxes through the computational cell. Extra terms are also added in the momentum equation, considering the drag, and in the turbulence source term in the k-epsilon equations. Full details of the implementation can be found elsewhere [6] [7].

3 Results

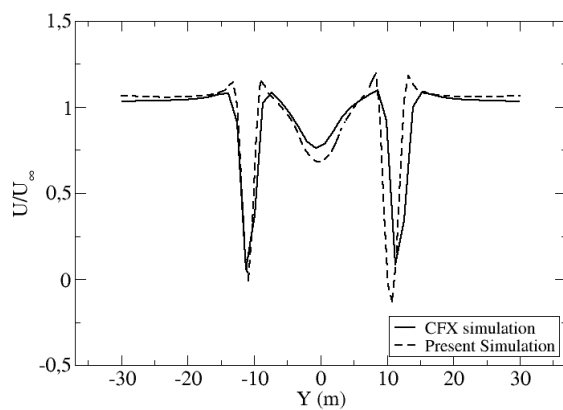
In the evaluation of the developed method to compute turbulent flows, in a first step we considered a cold flow simulation in a chemical process module presented in Figure 2. The velocity field have been calculated and numerical results were probed at 3 locations downstream the origin of the coordinate system and at 2 heights as listed below:

- Monitor Line 01: $x = 3$ m and $z = 5$ m.
- Monitor Line 02: $x = 3$ m and $z = 22$ m.
- Monitor Line 03: $x = 18$ m and $z = 5$ m.
- Monitor Line 04: $x = 18$ m and $z = 22$ m.
- Monitor Line 05: $x = 50$ m and $z = 5$ m.
- Monitor Line 06: $x = 50$ m and $z = 22$ m.

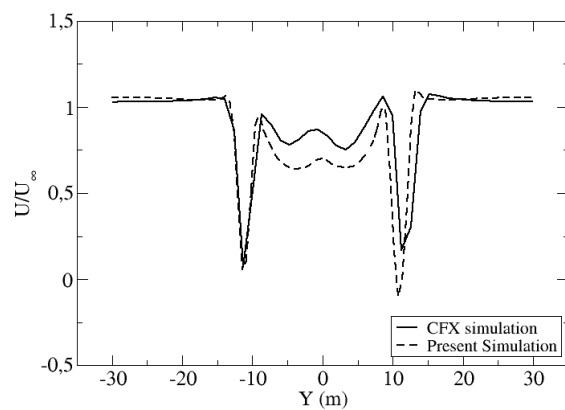
The numerical results from the numerical solution of the Navier-Stokes are shown in the plots as "present simulation". Since there is no experimental data for the cases considered the numerical findings obtained via the novel technique are compared with available CFD (Computational Fluid Dynamics) code, namely Ansys CFX. Comparison among the present simulation and the standard method for solution of the Navier-Stokes equations (Fig 4.a - Fig. 4f) shows good agreement.

As expected, the velocity drops significantly at regions behind the large equipment. Following the line transverse to the flow orientation shows a uniform flow where the normalised velocity is one.

At the regions inside the module the velocity drops due to the presence of obstacle dropping to values close to zero (Fig 4.a - Fig. 4e). At regions inside the module where the interference caused by the obstacle is less pronounced the velocity is reduced. However it does not reach value smaller than 50% of the bulk velocity. It is worth mentioning that it has also been observed regions in the module where the velocity increases due to the reduced area to the flow. This is in line with the principle of conservation of mass that ensures higher velocities at regions where the available area to the flow is reduced.



(a)



(b)

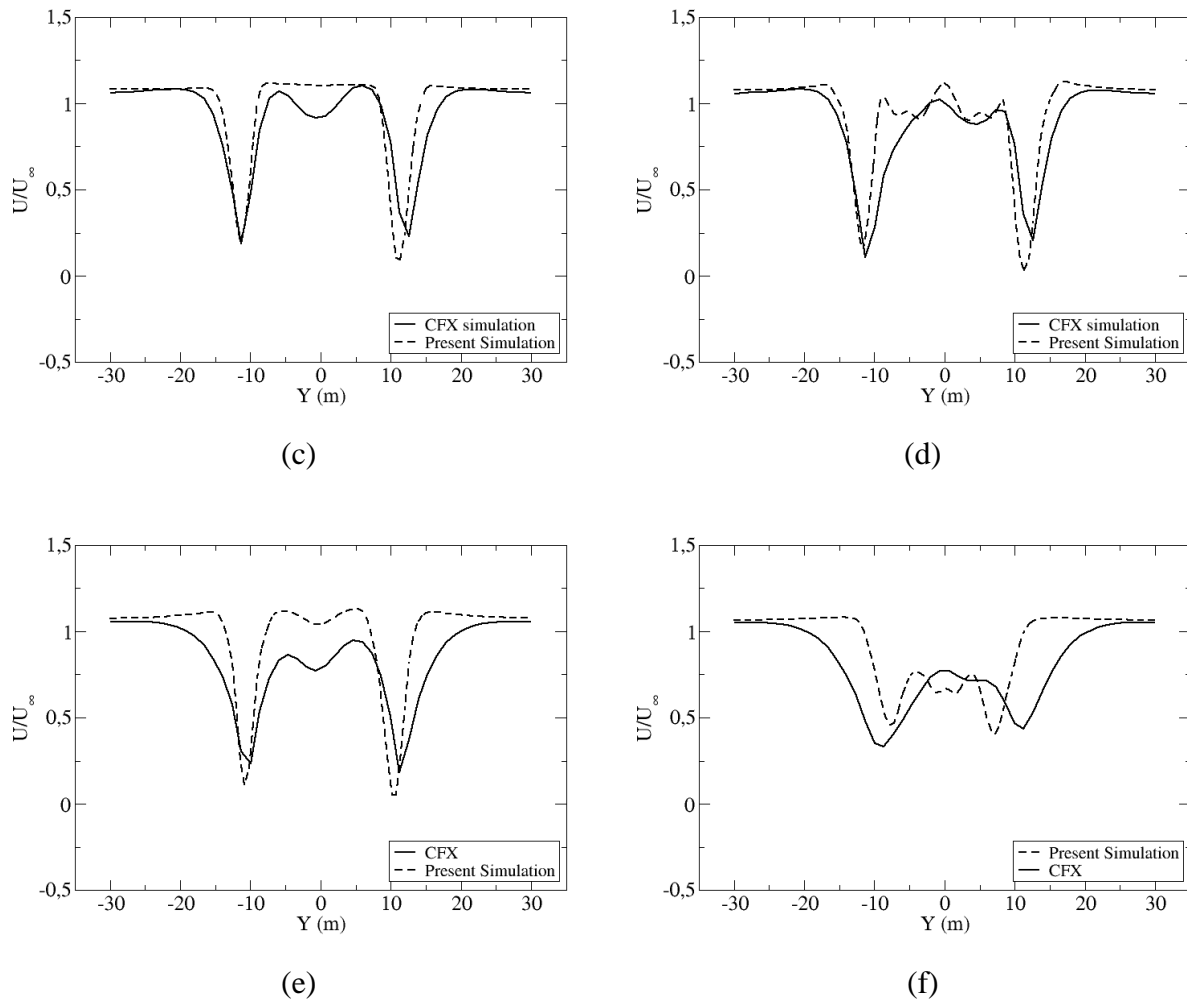


Figure 4: Velocity profile at six different locations inside a chemical process area (Figure 2): (a) monitor line 01, (b) monitor line 02, (c) monitor line 03, (d) monitor line 04, (e) monitor line 05, (f) monitor line 06. The numerical findings provided by the developed model presents good agreement with the standard CFD tool Ansys CFX [7].

We also investigate how the flamelet combustion model coupled with the collision distance algorithm reproduces turbulent reacting flows. In previous work, we have compared the numerical findings provided by the developed approach compare with experimental data for combustion scenarios in small scale geometry [6]. Here, two different combustion simulation were performed and the results were compared with the CFD commercial tool FLACS for gas explosion.

The first set of explosion analysis comprises a combustion chamber, measuring of 500 mm x 80 mm x 80 mm, filled with a flammable mixture of air and propane. Figure 5 presents the flame position into the chamber at different time steps. As expected for premixed combustion, the flame advances towards through the chamber in the reactant direction. The flame tip reaches the obstacle and its structure is changed to assume a finger shape. Similar behaviour is observed in both present and FLACS simulation.

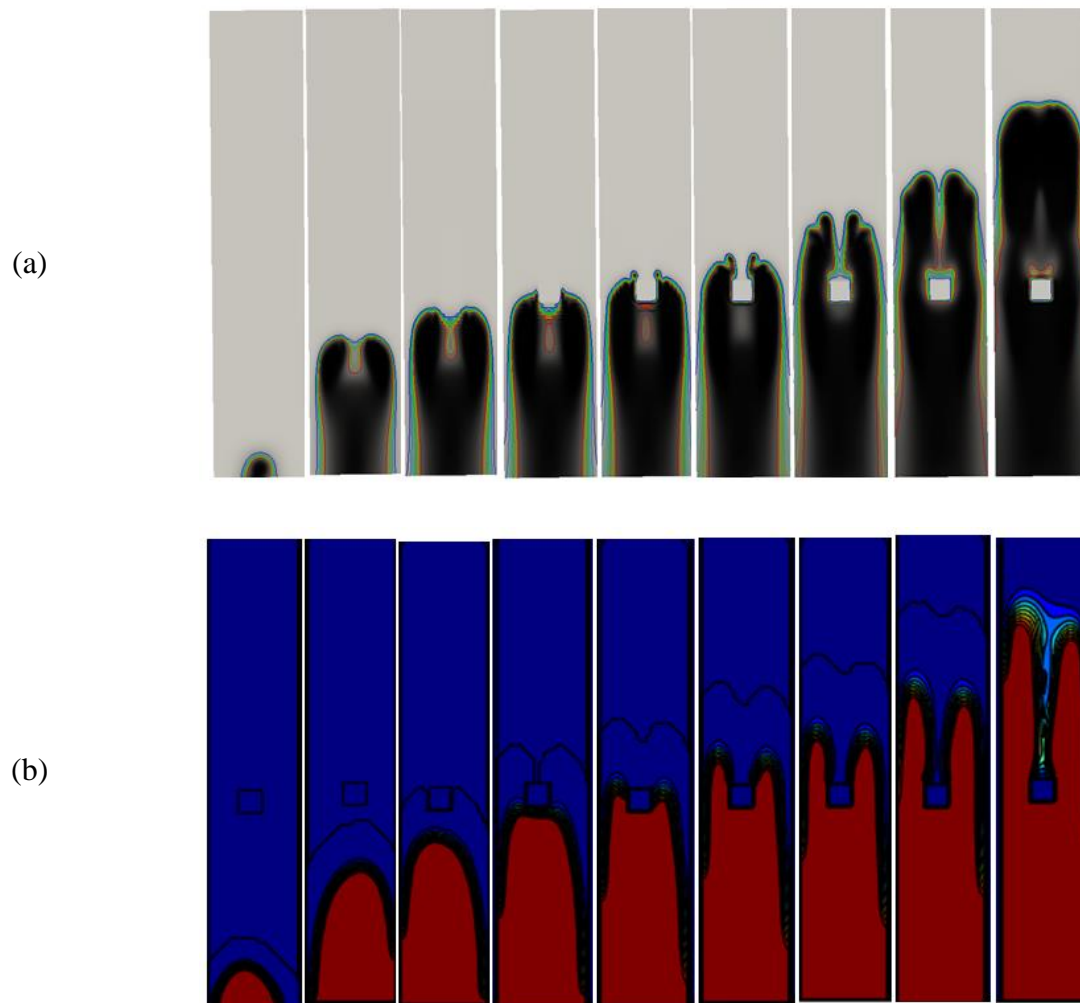


Figure 5: Flame position into a combustion chamber obstructed by a single square obstacle. The chamber was initially filled with a stoichiometric mixture of air and propane. The flame position is observed at different time steps considering numerical data provided by the developed approach (a) and FLACS CFD tool (b).

The second explosion analysis considers a real scale geometry with a chamber measuring 9 m x 4.5 m x 4.5 m. A set of 80 tubes with 0.5 m of diameter were placed into the chamber in order to make a congestion region. The chamber was filled with a stoichiometric mixture of methane and air. A ignition point was located in the beginning of the chamber and the flame behaviour was observed in different time steps after the ignition. Figure 6 presents the numerical results provided by the present approach and the FLACS CFD tool.

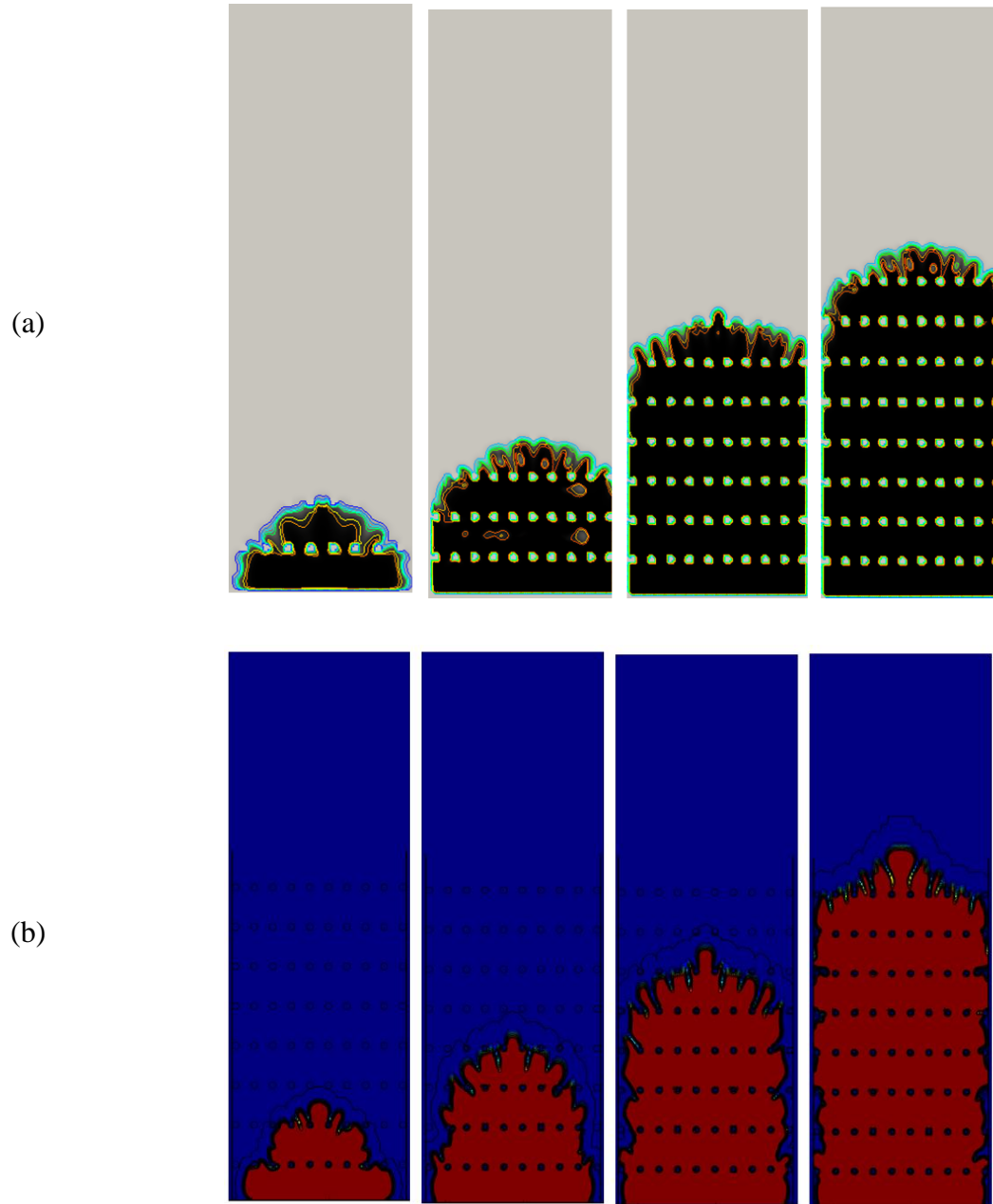


Figure 6: Progress variable advancement in a chamber congested by cylinders. The flame position is observed at different time steps considering numerical data provided by the developed approach (a) and FLACS CFD tool (b).

4 Conclusion and Future work

We have introduced a method for computation of turbulent flow in complex geometries. The new approach combines the finite volume method with Minkowski addition for correction of the flux through the area of the computational cell and its respective volume. The method is

formulated as a convex combination of the simplex (simplex makes reference to a point, a line or any convex region) searching for the nearest reference point (The supporting point can be understood as the furthest point in a given direction). To demonstrate the approach we considered a simple geometry represented by a cube. The level of complexity was raised and two additional geometrical models have been considered. The first complex model was a chemical processing module and the second complex problem dealt with a full chemical process plant assembled on the topside of a vessel. Both complex geometrical models comprise details of the geometry ranging from a few inches to a few metres. We can correctly identify and map all the details of the geometry associated with the original CAD model using the Minkowski sum. Moreover, we have solved the set of Navier-Stokes using the modified formulation of the finite volume method using the Minkowski sum and the numerical findings agree with the selected benchmarking. We found that the numerical results obtained by the proposed method are in accordance with numerical findings provided by standard commercial CFD tools, even where the geometrical model has been considered in the simulations.

5 References

1. Vianna, S. and Cant, R. (2010). Modified porosity approach and laminar flamelet modelling for advanced simulation of accidental explosion. *Loss Prevention in the Process Industries*. 23, 3–14.
2. Savill, A. and Solberg, T. (1994). Some improvements to pdr/k – ϵ model predictions for explosions in confined geometries. In *Proceedings ER-COFTA/IMA Confined Flow Dispersion Through Groups Obstacles*, Cambridge, UK, 227–247.
3. Forthergill, C., Chynowet, S., Roberts, P. and Packwood, A. (2003). Evaluation a cfd porous model for calculating ventilation in explosion hazard assessments. *Loss Prevention in the Process Industries*. v. 4, 16, 341–347.
4. Gilbert, E., Johnson, D. and Keerthi, S. (1988). “A fast procedure for computing the distance between complex objects in three-dimensional space”. *Robotics and Automation*. v. 2, 4, 193–203.
5. van den Bergen, G., Van, G. and Bergen, D. (1999). “A fast and robust gjk implementation for collision detection of convex objects”. *Journal of Graphics Tools*. 7–25.
6. Moreira, V.D., Ferreira, T.D., Santos, R.G., Vianna, S.S.V. (2015). A novel technique to obtain a mesh porosity based on the Gilbert-Johnson-Keerthi distance algorithm. 23rd ABCM International Congress of Mechanical Engineering. *Proceedings*, 1-9
7. Ferreira, T. D., Santos, R.G., Vianna, S.S.V. (2019). A coupled finite volume method and Gilbert–Johnson–Keerthi distance algorithm for computational fluid dynamics modelling. *Computer Methods in Applied Mechanics and Engineering*. 352, 417-436.
8. Ferreira, T.D and Vianna, S.S.V. (2019). The Gilbert Johnson Keerthi distance algorithm coupled with computational fluid dynamics applied to gas explosion simulation. *Process Safety and Environmental Protection*. 130, 209-220.