



UNIVERSIDADE ESTADUAL DE CAMPINAS  
Faculdade de Engenharia Química

TATIELE DALFIOR FERREIRA

DEVELOPMENT OF A 3D NAVIER STOKES SOLVER COMBINED  
WITH THE GILBERT-JOHNSON-KEERTHI DISTANCE  
ALGORITHM FOR SIMULATION OF TURBULENT REACTING  
FLOW IN COMPLEX GEOMETRIES

DESENVOLVIMENTO DE UM SOLVER NAVIER STOKES 3D  
COMBINADO COM O ALGORITMO DE DISTÂNCIA DE  
GILBERT-JOHNSON-KEERTHI PARA SIMULAÇÃO DE  
ESCOAMENTO TURBULENTO REATIVO EM GEOMETRIAS  
COMPLEXAS

CAMPINAS

2018

TATIELE DALFIOR FERREIRA

DEVELOPMENT OF A 3D NAVIER STOKES SOLVER COMBINED  
WITH THE GILBERT-JOHNSON-KEERTHI DISTANCE  
ALGORITHM FOR SIMULATION OF TURBULENT REACTING  
FLOW IN COMPLEX GEOMETRIES

DESENVOLVIMENTO DE UM SOLVER NAVIER STOKES 3D  
COMBINADO COM O ALGORITMO DE DISTÂNCIA DE  
GILBERT-JOHNSON-KEERTHI PARA SIMULAÇÃO DE  
ESCOAMENTO TURBULENTO REATIVO EM GEOMETRIAS  
COMPLEXAS

Thesis presented to the Faculty of Chemical Engineering of the University of Campinas in partial fulfillment of the requirements for the degree of Doctor in Chemical Engineering.

Tese apresentada à Faculdade de Engenharia Química da Universidade Estadual de Campinas como parte dos requisitos exigidos para obtenção do título de Doutora em Engenharia Química.

Supervisor: PROF. DR. SÁVIO SOUZA VENÂNCIO VIANNA

Co-supervisor: PROF. DR. ROGÉRIO GONÇALVES DOS SANTOS

ESTE EXEMPLAR CORRESPONDE À  
VERSÃO FINAL DA TESE DEFENDIDA  
PELA ALUNA TATIELE DALFIOR FER-  
REIRA, E ORIENTADA PELO PROF.  
DR. SÁVIO SOUZA VENÂNCIO VIANNA

CAMPINAS

2018

**Agência(s) de fomento e nº(s) de processo(s):** CNPq, 140812/20148

Ficha catalográfica  
Universidade Estadual de Campinas  
Biblioteca da Área de Engenharia e Arquitetura  
Rose Meire da Silva - CRB 8/5974

F413d Ferreira, Tatiele Dalfior, 1988-  
Development of a 3D Navier-Stokes solver combined with the Gilbert-Johnson-Keerthi Distance Algorithm for simulation of turbulent reacting flow in complex geometries / Tatiele Dalfior Ferreira. – Campinas, SP : [s.n.], 2018.

Orientador: Sávio Souza Venâncio Vianna.

Coorientador: Rogério Gonçalves dos Santos.

Tese (doutorado) – Universidade Estadual de Campinas, Faculdade de Engenharia Química.

1. ExploFluidodinâmica Computacional. I. Vianna, Sávio, Souza Venâncio, 1975-. II. Santos, Rogério Gonçalves, 1978-. III. Universidade Estadual de Campinas. Faculdade de Engenharia Química. IV. Título.

#### Informações para Biblioteca Digital

**Título em outro idioma:** Desenvolvimento de um solver Navier-Stokes 3D combinado com o algoritmo de distância de Gilbert-Johnson-Keerthi para simulação de escoamento turbulento em geometrias complexas

**Palavras-chave em inglês:**

Computational Fluid Dynamics

**Área de concentração:** Engenharia Química

**Titulação:** Doutora em Engenharia Química

**Banca examinadora:**

Sávio Souza Venâncio Vianna [Orientador]

Clovis Raimundo Maliska

João Lameu da Silva Jr

Luis Fernando Mercier Franco

Antônio Carlos Luz Lisbôa

**Data de defesa:** 03-10-2018

**Programa de Pós-Graduação:** Engenharia Química

Tese de Doutorado defendida por Tatiele Dalfior Ferreira e aprovada em 03 de outubro de 2018 pela banca examinadora constituída pelos doutores:

---

Prof. Dr. Sávio Souza Venâncio Vianna – Orientador  
FEQ/UNICAMP

---

Prof. Dr. Antonio Carlos Luz Lisbôa  
FEQ/UNICAMP

---

Prof. Dr. Luís Fernando Mercier Franco  
FEQ/UNICAMP

---

Dr. João Lameu da Silva Júnior  
IFSULDEMINAS

---

Dr. Clóvis Raimundo Maliska  
Simeq/UFSC

A Ata da defesa com as respectivas assinaturas dos membros encontra-se no SIGA/Sistema de Fluxo de Tese e na Secretaria do Programa da Unidade

*To my Parents.*

*Aos meus Pais.*

# Acknowledgements

First and foremost, I would like to thank God for His showers of blessings, for the wisdom and perseverance that He has been bestowed upon me during this project and throughout my life.

I would like to express my deep gratitude to my advisor Prof. Sávio S.V. Vianna for believing me and challenging me to develop this work. For his continuous support of my Ph.D study and related research, his patience, motivation, and knowledge. His guidance helped me in all the time and allowed me to grow as a research scientist.

My appreciation and sincere thanks also goes to Prof. Milton Mori, who provided me an opportunity to join their team in the PQGE Lab for the development of this work. His precious support helped me in my academic and professional life in the School of Chemical Engineering at Unicamp.

I would like to thank Prof. Rogerio Gonçalves, my co-Supervisor, for his support, knowledge and encouragement.

My deep and sincere gratitude to my fellow labmate Victor Dammato Moreira (*In Memoriam*). His wonderful work and research was essential in the development of this project.

I thank my friends in L4RIS4 and PQGE Lab for your friendship, support and cooperative in implementing this project.

I am grateful to the scientific interns of L4RIS4 Lab for all help in the tests performed during this work.

Thanks are due to the Brazilian Research Council CNPq for the financial support in this work.

I want to thank all of my family and friends who supported and incentive me to strive towards my goal.

Last but not the least, I am extremely grateful to my parents Luiz C. P. Ferreira and Geny D. Ferreira for their unconditional love, prayers, caring and sacrifices for educating and preparing me for my future. I am very much thankful to my sister

Gabriela D. Ferreira for always believing in me and for her love. Thank you my family!  
Your prayers for me were what sustained me thus far.

# Agradecimentos

Primeiramente, e mais importante, eu gostaria de agradecer a Deus por suas chuvas de bênçãos, pela sabedoria e perseverança que Ele me concedeu durante este projeto e durante toda a minha vida.

Gostaria de expressar minha profunda gratidão ao meu orientador Prof. Sávio S.V. Vianna por acreditar em mim e me desafiar a desenvolver esse trabalho. Por seu apoio contínuo aos meus estudos de e pesquisa de doutorado, sua paciência, motivação e conhecimento. Sua orientação me ajudou o tempo todo e permitiu que eu crescesse como pesquisadora.

Meu apreço e sinceros agradecimentos também ao Prof. Milton Mori, que me proporcionou a oportunidade de me juntar à sua equipe e fornecer o Laboratório PQGE para o desenvolvimento deste trabalho. Seu precioso suporte foi de extrema importância para minha vida acadêmica e profissional na Faculdade de Engenharia Química da Unicamp.

Gostaria de agradecer ao Prof. Rogério Gonçalves, meu co-orientador, pelo seu apoio, conhecimento e incentivo.

Minha profunda e sincera gratidão ao meu companheiro de laboratório Victor Damato Moreira (*em memória*). Seu maravilhoso trabalho e pesquisa foram essenciais no desenvolvimento deste projeto.

Agradeço aos meus amigos dos Laboratórios de Pesquisa L4RIS4 e do PQGE pela sua amizade, apoio e colaboração na implementação deste projeto.

Sou grata aos alunos de iniciação científica do Laboratório L4RIS4 por toda ajuda nos testes realizados durante este trabalho.

Agradecimentos são dados ao CNPq (Conselho Brasileiro de Apoio à Pesquisa) pelo apoio financeiro para realização deste trabalho

Agradeço a todos os meus amigos e familiares que me apoiaram e incentivaram a buscar meu objetivo.

Por último, mas não menos importante, sou extremamente grata aos meus pais Luiz C. P. Ferreira e Geny D. Ferreira pelo seu amor incondicional, orações, carinho e sacrifícios, por me educarem e me prepararem para o futuro. Sou muito grata a minha irmã



Gabriela D. Ferreira por sempre acreditar em mim e por seu amor. **Obrigada familia!**  
Suas orações foram o que me sustentaram até agora.

*"I do not pray for a lighter load, but for a strong back."*

*Phillips Brooks*

*"The first requisite for success is to develop the ability to focus and apply your mental and physical energies to the problem at hand - without growing weary."*

*Thomas A. Edison*

*"Eu não oro por cargas mais leves, mas por ombros mais fortes."*

*Phillips Brooks*

*"O primeiro requisito para o sucesso é a habilidade de aplicar incessantemente suas energias físicas e mentais a qualquer problema, sem se cansar."*

*Thomas A. Edison*

# Abstract

In this work we propose the development of a three-dimensional Navier-Stokes solver for analysis of turbulent reacting flows in complex geometries. Concepts of Porosity Distributed Resistance were coupled with Computational Fluid Dynamics (CFD) techniques and the Gilbert-Johnson-Keerthi distance algorithm was applied to describe geometrical models as porous media. The solver was based on an initial two dimensional compressible Euler solver and has been improved until becomes a three-dimensional solver capable to compute the Navier-Stokes equations. The numerical modelling was conducted within the framework of traditional Reynolds-Averaged Navier Stokes (RANS) approach and the turbulence closure model was addressed via Boussinesq formulation. The solver was customized to a specific class of turbulent reacting flow by modelling combustion process as gas explosion. The approach was demonstrated to handle complex geometries within feasible computational time. Numerical findings for simulation of non-reacting flows and reacting flows present the main features of the fluid flow and good agreement with experimental data was observed.

**Keywords:** Gas Explosion; Porosity Distributed Resistance; Computational Fluid Dynamics; Gilbert-Johnson-Keerthi algorithm.

# Resumo

Este trabalho tem como objetivo o desenvolvimento de um solver tridimensional de Navier-Stokes para análise de escoamentos turbulentos reativos em geometrias complexas. Os conceitos de Resistência Distribuída por Porosidade foram acoplados às técnicas de Fluidodinâmica Computacional (CFD) e o algoritmo de distância de Gilbert-Johnson-Keerthi foi aplicado para descrever modelos geométricos como meios porosos. O programa computacional foi desenvolvido com base em um solver Euler compressível e bidimensional e foi melhorado até se tornar um solver tridimensional capaz de resolver as equações de Navier-Stokes. A modelagem numérica foi conduzida dentro da abordagem tradicional de Reynolds-Averaged Navier Stokes (RANS) e o modelo de fechamento de turbulência foi abordado através da formulação de Boussinesq. O solver desenvolvido foi customizado para uma classe específica de escoamentos turbulentos reativos, modelando o processo de combustão como uma explosão de gás. Foi demonstrado que esta abordagem é capaz de manipular geometrias complexas dentro de um tempo computacional viável. Resultados numéricos para simulação de escoamento não-reativos e reativos apresentaram as principais características previstas para o escoamento de fluidos e foi observado uma boa concordância com dados experimentais disponíveis na literatura.

**Palavras-chaves:** Explosão de gases; Resistência Distribuída por Porosidade; Fluidodinâmica Computacional; Algoritmo de Gilbert-Johnson-Keerthi.

# List of Figures

Figure 1 – Examples of complex geometries . . . . .	23
Figure 2 – Regime diagram for turbulent premixed combustion . . . . .	31
Figure 3 – Fluctuations in a turbulent flow - example of a velocity field . . . . .	39
Figure 4 – Typical computational cell partially occupied by a solid object and the projected area in the cell faces . . . . .	52
Figure 5 – Steps of the porosity mesh calculation . . . . .	59
Figure 6 – Porosity recursion . . . . .	61
Figure 7 – Porosity applied to a simple oil platform model . . . . .	62
Figure 8 – General grid in finite volume method . . . . .	63
Figure 9 – The hexahedral computational cell in PFS code . . . . .	64
Figure 10 – Two-Dimensional Mesh Discretization . . . . .	64
Figure 11 – Subdivision of an hexahedral volume in pyramids . . . . .	65
Figure 12 – Main steps of a simulation using PFS code . . . . .	66
Figure 13 – Set of hexahedral computational cells surrounding and influencing a node in PFS code . . . . .	67
Figure 14 – Simulation of a subsonic flow in a three dimensional channel with a stricture in the center by solving Euler equation . . . . .	74
Figure 15 – Simulation of a supersonic flow in a three dimensional channel with a stricture in the center by solving Euler equation . . . . .	75
Figure 16 – Parametrisation of the cube .stl CAD file used as the bluff body in the fluid flow simulation using the 3D Euler solver. (a) Cube (CAD file) in the computational domain. (b) Uniform hexahedral cells of the compu- tational domain. (c) Porosity field calculated by the GJK algorithm os shown in the central plane of the computational domain . . . . .	75
Figure 17 – Simulation of a flow over a parametrized squared geometry by using the 3D Euler solver . . . . .	77
Figure 18 – Mesh test applied to a simulation of a flow over the parametrised squared geometry by using the 3D Euler solver . . . . .	78
Figure 19 – CFL test applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver . . . . .	79
Figure 20 – Numerical test with different values of smooth factor applied to a sim- ulation of a flow over a parametrised squared geometry by using the 3D Euler solver . . . . .	79
Figure 21 – Qualitative analysis of different values of smooth factor applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver . . . . .	80

Figure 22 – Time mean velocity profile at different distances upstream and downstream the rectangular box . . . . .	81
Figure 23 – Time mean velocity profile at the centreline downstream of the rectangular box . . . . .	82
Figure 24 – Schematic representation of the simulation of a turbulent flow over a backward-facing step . . . . .	83
Figure 25 – Time mean velocity profile at the distance $x/H = 7$ downstream the step for three different sizes of mesh. . . . .	84
Figure 26 – Simulation of a turbulent flow over a backward-facing step - velocity field	84
Figure 27 – Non-dimensional velocity profile of a turbulent flow over a backward-facing step . . . . .	85
Figure 28 – Turbulence intensity of a turbulent flow over a backward-facing step . .	86
Figure 29 – Schematic representation of the geometry used in the study of a flow around a rectangular cylinder . . . . .	87
Figure 30 – Time mean velocity profile along the centreline of a square cylinder for three different sizes of mesh. . . . .	88
Figure 31 – Velocity field of a turbulent flow by considering the geometry of a rectangular cylinder as a porous media . . . . .	89
Figure 32 – Comparison between PFS simulation results and experimental data along the centreline of a squared cylinder . . . . .	90
Figure 33 – Schematic representation of the simulation of a turbulent flow over a mounted squared cylinder . . . . .	90
Figure 34 – Velocity field of a turbulent flow over a mounted cylinder in a longitudinal symmetry plane . . . . .	91
Figure 35 – Velocity field of a turbulent flow over a mounted cylinder near the channel floor . . . . .	91
Figure 36 – Velocity and a turbulence intensity profiles in a symetry plane at different distances from the begining of the mounted squared cylinder . .	92
Figure 37 – Schematic representation of the simulation of flow around cylinders - CMR tests . . . . .	93
Figure 38 – Velocity flow field for CMR experiments . . . . .	93
Figure 39 – Comparison between PFS simulation and experimental data (ARNTZEN, 1998) considering the mean velocity profile and the turbulent velocity fluctuation at the region downstream the considered geometries . . . .	94
Figure 40 – Schematic representation of the simulation of turbulent cross-flow in a staggered tube bundle . . . . .	95
Figure 41 – Porous mesh obtained for a staggered tube bundle . . . . .	96
Figure 42 – Scalar velocity field obtained as result of the PFS simulation of turbulent cross-flow in a staggered tube bundle . . . . .	96

Figure 43 – Comparison of the simulation of a turbulent cross-flow in a staggered tube bundle by the developed PFS tool and ANSYS-CFX. Velocity profiles were taken at four different regions of the domain . . . . .	97
Figure 44 – Idealized offshore module adopted for flow simulation in a complex geometry. . . . .	98
Figure 45 – Ventilation in a offshore module provided by PFS simulation . . . . .	98
Figure 46 – Comparison of the simulation of air flow in a offshore module provided by the developed PFS tool and ANSYS-CFX . . . . .	100
Figure 47 – Schematic representation of a preliminary combustion simulation . . .	101
Figure 48 – Three-dimensional flame countour ( $c=0.4$ ) and vector field at 4.6ms after ignition . . . . .	102
Figure 49 – Preliminary explosion simulation using PFS code: reaction progress variable, fuel fraction, pressure and temperature field at 4.6ms after ignition . . . . .	103
Figure 50 – Preliminary explosion simulation using PFS code: non-dimensional numbers at 4.6ms after ignition . . . . .	104
Figure 51 – Sensitive analysis of the initial laminar combustion phase . . . . .	105
Figure 52 – Analysis of the influence of adjustable variable $c_L$ from BML combustion model on flame propagation velocity. . . . .	105
Figure 53 – Schematic representation of a combustion chamber obstructed by a single obstacle . . . . .	106
Figure 54 – Time histories of flame position and overpressure of a combustion chamber obstructed by a single obstacle . . . . .	107
Figure 55 – Flame position into a combustion chamber obstructed by a single obstacle at different time instants after ignition starts . . . . .	108
Figure 56 – Schematic representation of a combustion chamber with three obstacles	109
Figure 57 – Time sequance of the flame position at the central plane of the chamber	109
Figure 58 – Comparison between numerical simulation and experimental data for the flame location inside the chamber . . . . .	110
Figure 59 – Experimental data and corresponding numerical predictions for the flame speed profile along the axial distance from the igitation point . .	111
Figure 60 – Time pressure history at the bottom end of the chamber - comparison between numerical results and experimental data . . . . .	112
Figure 61 – Dimensionless <i>Damköler</i> and Karlovitz numbers at the flame front region for different time instants . . . . .	113
Figure 62 – Schematic representation of a combustion chamber with three different obstacles blockage ratios . . . . .	114
Figure 63 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with $BR = 0.3$ . . . . .	115



Figure 64 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with BR = 0.5 . . . . .	116
Figure 65 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with BR = 0.7 . . . . .	116
Figure 66 – Vector velocity field generated by the obstacle for the blockage ratio equal to 0.5 . . . . .	117
Figure 67 – Logical sequence and relationship between the different subroutines of the developed PFS code . . . . .	126
Figure 68 – Routines of the developed code and the main link between them . . . . .	131
Figure 69 – Parametrization of a industrial module as porous model by using the GJK algorithm and its corresponding flow solution for an onshore site . . . . .	132
Figure 70 – Parametrization of process deck and associated flow solution. . . . .	133
Figure 71 – Parametrization of a gas plant including details of the pipe-rack. Lower plot shows the streamlines for the velocity field. . . . .	134
Figure 72 – Parametrization of a gas plant including details of the pipe-rack. Lower plot shows a flammable cloud (green region) inside the process area. . . . .	135
Figure 73 – Explosion simulation in a process area. The pressure wave is shown at different times after the ignition: (a) t = 35ms; (b) t = 70ms; (c) t = 105ms; (d) t = 180ms. . . . .	136

# List of Tables

Table 1	– Constant values for the standard $k$ - $\varepsilon$ turbulence model . . . . .	44
Table 2	– Constant values for the laminar flame speed correlation . . . . .	50
Table 3	– Main set up features for PFS simulations. . . . .	72
Table 4	– Solver parameters used in the simulation of a flow over a parametrized squared geometry by using the 3D Euler solver . . . . .	76
Table 5	– Solver parameters used in the simulation of a flow over a backward facing step by solving the set of Navier-Stokes equations . . . . .	83
Table 6	– Main set up features for a preliminary combustion simulation. . . . .	102

# Contents

<b>1</b>	<b>Introduction</b>	<b>22</b>
1.1	General Goal	24
1.2	Specific Goals	24
1.3	This Doctor Thesis	24
<b>2</b>	<b>Gas Explosion</b>	<b>26</b>
2.1	Combustion and Explosion	26
2.1.1	Gas Explosion Phenomenon	27
2.1.1.1	Confined Gas Explosions	27
2.1.1.2	Partly Confined Explosions	27
2.1.1.3	Unconfined Gas Explosions	28
2.1.1.4	Deflagration and Detonation	28
2.1.2	Combustion modelling for gas explosion	28
2.1.2.1	Premixed Flames	28
2.1.2.2	Laminar Premixed Flames	28
2.1.2.3	Turbulent premixed flames	29
2.1.2.4	Dimensionless numbers in Turbulent Premixed Flames and Regimes Diagram	30
2.1.3	Turbulent premixed flames modelling	32
2.1.3.1	Eddy Break-Up Model (EBU)	32
2.1.3.2	Flamelet Model	32
2.2	Models to describe the gas explosion phenomena	32
2.2.1	Empirical and Phenomenological models	33
2.2.2	CFD models	33
2.2.2.1	CFD Codes for Gas Explosion	34
<b>3</b>	<b>RANS Numerical Modelling for Turbulent Premixed Combustion</b>	<b>36</b>
3.1	Mass and momentum equations	36
3.1.1	The mass conservation equation	36
3.1.2	The momentum conservation equation	37
3.2	Elementary Description of turbulence	37
3.3	RANS turbulence modelling	39
3.3.1	Eddy Viscosity turbulence models	41
3.3.1.1	The $k$ - $\omega$ turbulence model	42
3.3.2	The $k$ - $\varepsilon$ turbulence model	43
3.3.2.1	Turbulence close to walls	44
3.4	Favre averaging	46
3.5	Numerical Combustion Modeling	46

3.5.1	Turbulent premixed combustion modelling by Bray-Moss-Libby . . .	47
3.5.1.1	Thermodynamics and Equation of State . . . . .	47
3.5.1.2	Reaction rate modelling . . . . .	48
3.5.2	Initial phase laminar combustion . . . . .	49
3.6	The Porosity Distributed Resistance Method (PDR) . . . . .	51
3.6.1	PDR Formulation . . . . .	51
3.6.2	Modified Conservation Equations by Using the Porosity Distributed Resistance (PDR) Model . . . . .	52
3.6.2.1	Additional resistance to the flow . . . . .	53
3.6.2.2	Turbulence production due to resistance . . . . .	53
<b>4</b>	<b>Porosity Flow Solver (PFS) . . . . .</b>	<b>55</b>
4.1	Structure of PFS . . . . .	56
4.1.1	Geometry . . . . .	56
4.1.1.1	Fundamentals of the GJK Algorithm . . . . .	56
4.1.1.2	Porosity Mesh Calculation . . . . .	58
4.1.2	Preprocessing . . . . .	62
4.1.2.1	Variable Storage . . . . .	62
4.1.2.2	PFS Mesh . . . . .	63
4.1.3	PFS Solver . . . . .	65
4.1.4	Postprocessing . . . . .	66
4.2	Discretisation Process . . . . .	67
4.2.1	Discretisation of the Conservation Equations . . . . .	67
4.2.1.1	The Unsteady Term . . . . .	68
4.2.1.2	The Convection Term . . . . .	68
4.2.1.3	The Diffusion Term . . . . .	68
4.2.1.4	The Source Term . . . . .	69
4.2.1.5	The Complete Discretised Conservation Equation . . . . .	69
4.2.2	The time Advancement . . . . .	69
4.2.2.1	Time Step . . . . .	70
4.2.2.2	Iterative Convergence . . . . .	70
4.2.3	Artificial Viscosity/Dissipation . . . . .	70
4.3	Boundary Conditions . . . . .	71
4.3.1	Initial Condition . . . . .	71
4.3.1.1	Ignition . . . . .	71
4.4	Set up and solver parameters . . . . .	72
<b>5</b>	<b>Case Studies for non-reacting flows . . . . .</b>	<b>73</b>
5.1	Verification Procedure . . . . .	73
5.1.1	3D Euler simulations . . . . .	73
5.1.2	Verification of the 3D Euler code with a parametrised geometry . . .	74

5.1.3	Simulation Parameters Analysis . . . . .	76
5.1.3.1	Mesh Test . . . . .	77
5.1.3.2	CFL Analysis . . . . .	78
5.1.3.3	Smooth Factor Analysis . . . . .	79
5.1.4	Verification of the 3D Navier-Stokes code . . . . .	81
5.2	Analysis of turbulent flows in simplified geometries . . . . .	82
5.2.1	Flow over a backward facing step . . . . .	82
5.2.2	Flow past a rectangular cylinder . . . . .	86
5.2.3	Flow over a mounted rectangular cylinder . . . . .	89
5.2.4	Flow around cylinders . . . . .	91
5.3	Analysis of turbulent flows in complex geometries . . . . .	95
5.3.1	Turbulent cross-flow in a staggered tube bundle . . . . .	95
5.3.2	Flow in an offshore module . . . . .	97
<b>6</b>	<b>Case Studies of reacting flows: Gas Explosion . . . . .</b>	<b>101</b>
6.1	Preliminary combustion test . . . . .	101
6.1.1	Sensitive tests of combustion model . . . . .	104
6.2	Case studies of combustion flows . . . . .	106
6.2.1	Study of a turbulent premixed flame past over an obstacle . . . . .	106
6.2.2	Analysis of a turbulent premixed flame accelerated by a sequence of obstacles . . . . .	108
6.2.3	Study of turbulent combustion in a chamber with different obstacles blockage ratio (OBR) . . . . .	113
<b>7</b>	<b>Conclusions and Future Work . . . . .</b>	<b>118</b>
7.1	Conclusions . . . . .	118
7.2	Future Work . . . . .	119
	<b>Bibliography . . . . .</b>	<b>121</b>

# 1 Introduction

Accidents involving fire and explosions represent a considerable hazard in industrial plants, particularly in oil and gas industry due to the presence of installations with congested areas and large quantities of flammable materials. Therefore, gas explosion might constitute a potential hazard for drilling, production and processing activities.

The release of flammable substances is the first step in the accidental chain of events. The next step is the generation and ignition of flammable clouds, resulting in explosions, which may in turn cause damage to the process plant (and in adjacent process areas), financial and human losses (ECKHOFF, 2005).

The prevention of gas explosion and the mitigation of its effects are crucial to ensure safety. The gas explosion hazard assessment can be very helpful to improve the design of the existing or new installations and characterise the extent of the hazard.

Making realistic predictions of the effects of an accidental explosion is a difficult task due to the complexity of the physical and chemical processes involved. There are limitations to experimental data available and full-scale experimental tests are often impracticable or prohibitively expensive (CANT, 2007). Thus, theoretical modelling and simulation have been employed to assess gas explosion hazards.

The numerical simulation using Computational Fluid Dynamics (CFD) is considered the best available approach when modelling gas explosion. The main strategy of CFD is to replace a continuous domain by a discrete computational domain using a grid where the differential equations that govern the flow are solved by using a numerical method.

In this context, the finite volume method (FVM) is the most popular numerical method for modelling of fluid flow transport problems as gas explosion. Following the FVM procedure, the computational domain of a given problem is discretised in a mesh of finite volumes where the conservation equations are solved.

Although the conventional finite volume method is well established and allows prediction of complex flows, a great deal of engineering problems requires a customised numerical technique. Thus, different approaches have been developed over the recent years to improve FVM formulation and bridge the gap between physical phenomena and numerical methods. In gas explosion analysis, as in a wide range of transport process, the geometrical model ranges from small length scales (a few inches) to large length scales. Such condition poses an additional challenge once details of the geometry are required for proper solution of the problem. In these cases, the mesh procedure requires a very fine

resolution and extensive computational effort to handle the complex geometry details. Figure 1 shows examples of complex geometries, which are characterized by the presence of different object scales and congestion.



Figure 1 – Examples of complex geometries with the presence of different object scales and congestion.

To model complex geometries and assess the great level of geometrical details, sub-grid models and Porosity Distributed Resistance (PDR) method have been applied in CFD simulations. The PDR method stands out as no mesh refinement is required around small scale geometry. By using this method, the large-scale geometry is fully represented and resolved, while the small-scale objects are taken together locally and approximated in terms of their effective porosity and resistance to the flow.

The Porosity Distributed Resistance (PDR) method is widely used for numerical simulations of gas explosion process and has been successful within the constraints of its modelling approximations.

This work presents a new procedure to obtain the porous mesh that takes into account all geometrical objects in the core of CFD (Computational Fluid Dynamics) simulations. The method is based on Gilbert – Johnson – Keerthi (GJK) distance algorithm (collision algorithm). The original formulation of the GJK algorithm is in accordance with the work reported by Gilbert, Johnson and Keerthi (GILBERT *et al.*, 1988), which was formulated to check collision between convex objects. The GJK algorithm is applied to check the collision between an element of the computational mesh and the primitive geometry. As a result of porosity calculation, the area and volume of each mesh element are calculated taking into account the availability for fluid flow. The developed method is applied for simulation of gas explosion in complex geometries.

## 1.1 General Goal

The present work aims at developing a three-dimensional Navier-Stokes solver for prediction of gas explosion in complex geometries by using the Gilbert - Johnson - Keerthi distance algorithm to parametrize complex geometries as porous media.

## 1.2 Specific Goals

The specific goals of this work are:

- Create an initial three-dimensional Euler solver and verify how the flux over the boundaries of the computational cells are predicted by the method proposed in the framework of this research;
- Apply the solver to calculate flow fields by using parametrised geometries by the GJK algorithm;
- Couple the viscous effects and the extra resistance and turbulence models with the amended fluxes in the FVM formulation;
- Evaluate the Damköhler hypothesis for reacting flows under the influence of GJK parametrisation;
- Verify the code implementation and validate the proposed model against a set of experimental data for cold and reacting flows.

## 1.3 This Doctor Thesis

The thesis is organised as follows:

In Chapter 1 a brief introduction is presented highlighting the motivations and the aim of this study.

The literature review is covered along the thesis, in Chapters 2 and 3.

Chapter 2 includes the main phenomena associated with gas explosion and the factors that influence its behaviour. The models that are used for gas explosion analysis are also discussed here.

Chapter 3 presents the governing equations that must be solved to describe the combustion process. The turbulence and combustion models applied in this work are also introduced.



In Chapter 4 the structure of the developed solver is discussed. The geometry and the mesh discretisation process as well as the solver architecture are fully explored in this part of the document.

Chapter 5 shows the numerical results using the developed solver. The code was initially tested for simulation of non-reacting flows. A sensitive analysis of the main simulation parameters is also presented.

Chapter 6 covers the simulation findings for turbulent reacting flows. It addresses different case studies.

Conclusions and future work are drawn in Chapter 7.

## 2 Gas Explosion

An explosion comprises a sudden increase in volume associated with an increase in temperature and pressure (ECKHOFF, 2005). During the explosion, large quantities of energy are released, causing considerable damage.

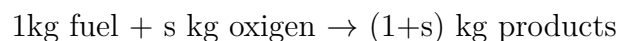
Due to highly destructive power of a gas explosion, the risk analysis from such accident is of paramount importance as far as process safety is concerned.

The physical description of the explosion phenomena and its mathematical modelling is presented in this chapter.

### 2.1 Combustion and Explosion

Combustion is defined as a reaction process in which a fuel is oxidized (usually by air) and involves heat release and often light emission. The products from a complete combustion of a hydrocarbon fuel are mainly water (vapour) and carbon dioxide.

Concerning gas explosion modelling, combustion is often treated as a single step irreversible chemical reaction with finite reaction rate (HJERTAGER, 1989). According to this concept, the reaction scheme may be written as:



where  $s$  is the stoichiometric oxygen requirement to burn 1 kg of fuel.

The combustion process between air and a gaseous fuel can occur in two different ways: non-premixed and premixed combustion. In the first case, fuel and oxygen are mixed during the combustion process. It occurs in many industrial systems, for safety reasons. Non-premixed combustion is applied for example in furnaces, diesel engines or gas turbines.

The second case comprises the situation where fuel and air are premixed and the fuel concentration must be within the flammability limits. Such mixing should be completed until the molecular level before combustion takes place. In premixed combustion the fuel burns faster when compared with non-premixed process.

In accidental gas explosion, the gas cloud is formed from the mixture between air and the gas released. If such mixture is within the flammability limits, it can ignite yielding an explosion. Hence, the phenomenon is frequently modelled based on the

concepts of premixed combustion. In the current research, the same approach has been adopted.

### 2.1.1 Gas Explosion Phenomenon

During the gas explosion, a strong positive feedback mechanism occurs. The flame is accelerated and when the flammable mixture of fuel and air is burning, the temperature increases and the gases expand. As consequence, the unburned gas is pushed ahead of the flame and a turbulent flow field is generated. The turbulence enhances the mixing and the reaction rate is increased. The combustion process is therefore also enhanced and a higher speed flow is produced, a higher turbulent field is generated and the process repeats.

The consequences of the gas explosion depends on a number of factors, such as maximum pressure, duration of shock wave, interaction with structures, etc. These factors, in turn, depend on a number of variables (LEA; LEDIN, 2002):

- The gas cloud characteristics: type of fuel, fuel concentration, and gas cloud size;
- The ignition point: location and strength of the ignition source;
- The geometrical characteristics of the process area: number, size, shape and location of obstacles;
- The degree of confinement.

The last parameter (the degree of confinement) has been used to classify the gas explosions as confined, partially confined, and unconfined explosions.

#### 2.1.1.1 Confined Gas Explosions

Confined explosions are also known as internal explosions. Explosions within tanks, process equipment, pipes, in closed rooms and in underground installations are classified as confined. An important characteristic of this kind of explosion is that even if the flame speed is slow, the overpressure can be high and the consequences can be severe.

#### 2.1.1.2 Partly Confined Explosions

Partly confined gas explosions occurs in installations that are partly opened, such as compression rooms and explosion modules. The building walls will confine the explosions and the pressure can be relieved through the open areas (as vents and flexible walls).

### 2.1.1.3 Unconfined Gas Explosions

The explosions that occur in open areas, such as large process plants, are understood as unconfined. Generally, this type of explosion produces small overpressure. However, partly confined and obstructed local areas may be present in process plants and in these cases high pressures can occur. Another important consideration is that even if the area is unconfined, there have been evidence that a detonation mode may occur yielding high overpressure values.

### 2.1.1.4 Deflagration and Detonation

The overpressure generated by the combustion wave depends on how fast the flame propagates and how the pressure can expand away from the gas cloud. When a cloud is ignited the flame can propagate in two different manners: deflagration and detonation (BJERKETVEDT *et al.*, 1997).

The deflagration occurs when the combustion wave propagates at subsonic velocity relative to unburned gas ahead of the flame. It is the most common mode of flame propagation in accidental gas explosions.

The detonation is defined as a combustion wave propagating at sonic velocity relative to the unburned gas. It can also be described as a shock wave immediately followed by a flame (BJERKETVEDT *et al.*, 1997). The overpressure generated by a detonation is much higher than a deflagration.

## 2.1.2 Combustion modelling for gas explosion

### 2.1.2.1 Premixed Flames

The most important behaviour characteristic of premixed flames is the "propagation". A premixed flame moves spontaneously in a normal direction to itself to consume the available reactant mixture. This is in contrast to a non-premixed flame which remain attached to the stoichiometric surface between fuel and oxidizer and cannot propagate (CANT; MASTORAKOS, 2008).

The rate of propagation in premixed flames quantifies the rate at which the flame can process reactants into products and the rate of heat release. According to this parameter, premixed flames can be classified as laminar or turbulent.

### 2.1.2.2 Laminar Premixed Flames

In laminar premixed flames, the propagation rate is specified in terms of the velocity of advance of the flame relative to reactants, also known as "laminar burning ve-

locity"  $u_L$  (CANT; MASTORAKOS, 2008). Considering a planar, unstretched, adiabatic laminar flame,  $u_L$  depends on only the thermochemistry of reactant mixture:

$$u_L = u_L (p, T_R, Y_1, Y_2, \dots, Y_N)$$

where  $p$  is the pressure,  $T_R$  is the reactant temperature and  $Y_\eta$  is the mass fraction of the chemical species.

A laminar flame presents the following main features:

- The preheat zone - where the temperature of reactants is raised by heat conducted forwards from the reaction zone.
- The reaction zone - where most of chemical reaction occurs. This is a very thin zone and the reaction rates are fast due to the high temperature and the presence of many reactive radical species.
- The equilibrium zone - where most heat release occurs. The temperature is high and chemical equilibrium is achieved.

Different theoretical approaches can be used for laminar flames, specially to understand their structure and the instabilities that can be developed on its front (POINSOT; VEYNANTE, 2005). However, in most studies, laminar premixed flames are applied in turbulent combustion modelling as the elementary step of turbulent flames. Considering numerical techniques, computing laminar premixed flames is the first step towards more complex configurations.

### 2.1.2.3 Turbulent premixed flames

When the turbulent flow enters a flame front, the laminar flame is replaced by a regime where turbulence and combustion interact. Eddies of different length scales act to wrinkle the planar flame surface, leading to the increasing of the flame surface area and hence to faster propagation. Some eddies may perturb the local internal structure of the flame and cause changes within the reactive-difusive balance that sustains the propagation. This may lead to a reduced local burning rate and slower propagation. In some cases, the turbulence may be strong enough to cause local extinction of the flame.

As described for laminar premixed flames, a turbulent burning velocity can be defined for turbulent premixed flames. Here, the turbulent burning velocity comprises the velocity of advance of the turbulent flame relative to the reactants, in the normal direction to itself and towards to reactants. The turbulent burning velocity is not a purely thermochemical quantity and it depends on the turbulence properties as the turbulence velocity fluctuation magnitude and length scale.

#### 2.1.2.4 Dimensionless numbers in Turbulent Premixed Flames and Regimes Diagram

In different fluid flow analysis, it is often helpful to describe flow features in terms of dimensionless numbers. To characterise turbulent premixed flames, a set of three main dimensionless numbers are applied. Such numbers are listed as follow:

- Turbulent Reynolds number

The Turbulent Reynolds number describes the ratio between the convective flux of momentum and the diffusive flux of momentum, and is obtained by:

$$Re_t = \frac{uL_{turb}}{\nu} \quad (2.1)$$

where  $u$  is the flame propagation velocity,  $L_{turb}$  is the integral turbulent length scale and  $\nu$  is the kinematic viscosity (diffusivity of momentum)

- Damköhler number

The Damköhler number describes the ratio between the turbulence large-eddy turn-over time-scale and the chemical time-scale representative of the laminar flame, so:

$$Da = \frac{L_{turb}u_L}{\delta_L u} \quad (2.2)$$

where  $\delta_L$  is the laminar flame thickness.

- Karlovitz number

The Karlovitz number describes the ratio between the laminar-flame chemical time-scale and the time scale of turbulent straining. Mathematically,  $Ka$  can be obtained by relation  $1/Da$ , or:

$$Ka = \frac{\delta_L u}{u_L \lambda_T} \quad (2.3)$$

where  $\lambda_T$  is the Taylor micro-scale <sup>1</sup>.

Turbulent premixed flames behave on different ways depending on turbulence intensity and length scale, chemical properties of reactants and products and the intensity

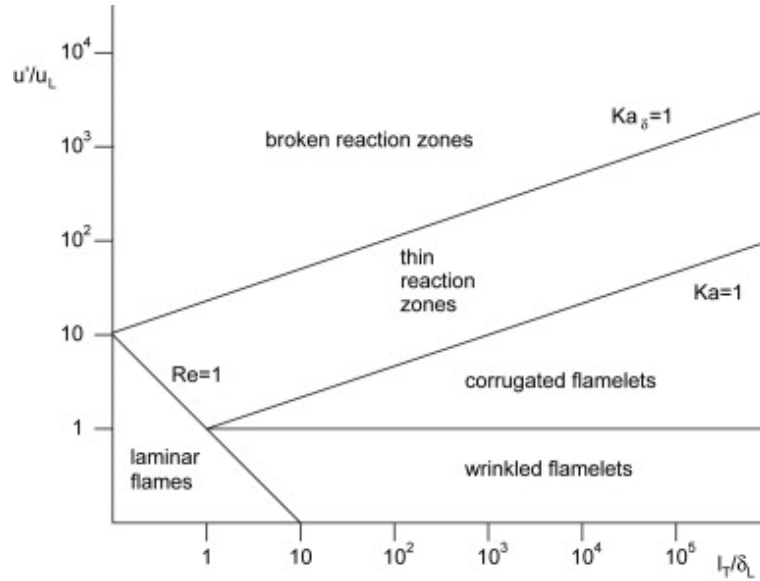


Figure 2 – Regime diagram for turbulent premixed combustion (CANT; MASTORAKOS, 2008). The turbulence length scale normalised by the laminar flame thickness are presented in axis "x" and is plotted against the turbulence intensity normalised by the laminar flame speed in axis "y".

of flame interaction. Such behaviours can be summarised in a regime diagram called Borghi diagram presented in Figure 2.

By analysing the Borghi diagram, the flame structure is divided into five main regions based on the ratio of the size of turbulent eddies to the laminar flame thickness on the x-axis  $L_T/\delta_L$ , and the ratio of turbulent velocity fluctuations to the laminar flame speed on the y-axis  $u'/u_L$ . The five regions are sub-divided by lines representing values of the three dimensionless numbers: the turbulent Reynolds number ( $Re_T$ ), the Damköhler number (Da) and Karlovitz number (Ka).

Following the diagram (Figure 2), the laminar flames are present where the turbulent Reynolds number is low ( $Re_T < 1$ ). The other four regions apply to turbulent flames. The diagram region where Reynolds and Damköhler numbers are high and Karlovitz number is low indicates moderately strong turbulence with fast chemistry and low turbulence strain rates, and comprises the known *laminar flamelet regime*. The laminar flamelets can exist in two subdivided regions of Borghi diagram: *wrinkled flamelet regime* (for values of  $u'/u_L$  less than a unit) and *corrugated flamelet regime* (where the turbulence intensity is higher). For values of Karlovitz number greater than a unity, the flame reaction zone is thin and presents a similar laminar structure, so this region is called *thin reaction zone regime*. The last flame regime is known as *broken reaction zone* and lies in the region of the diagram where the Karlovitz number based on the reaction zone

<sup>1</sup> Length scale where the effects of viscosity are relevant, as opposite to length scales where viscous effects can be neglected

thickness becomes greater than a unity, which indicates that local flame extinction can occur.

### 2.1.3 Turbulent premixed flames modelling

#### 2.1.3.1 Eddy Break-Up Model (EBU)

The Eddy Break-Up model (EBU) is one of the simplest and widely-used models for mean turbulent reaction rate in turbulent premixed flames. The model is based on the assumption that the chemistry is fast compared to the rate of turbulent transport, so the mean reaction rate is controlled by the rate at which the turbulence can bring fresh reactants into contact to hot products.

Combustion models based on EBU concepts are very simple to implement and are useful to represent flame behaviour for the particular cases of fast-chemistry limit of turbulent premixed flames.

#### 2.1.3.2 Flamelet Model

In some turbulent premixed flames, the chemical reaction rate cannot be considered infinitely fast. To some extent, for the flames which are not severely disturbed by the turbulence, the local structure of the flame at each point of the flame front is supposed to be similar to a laminar flamelet.

The flamelet approach is based on the description of the turbulent flame as a collection of laminar flame elements embedded in a turbulent flow and interacting with it. In the flamelet theory, the laminar flamelet consists in a thin and wrinkled reaction surface in which the combustion chemistry, as well as the heat release and mass transfer, takes place (CANT; MASTORAKOS, 2008). This interface propagates just like a strained and curved laminar flame.

The flamelet modelling has become well accepted for premixed flames and it has proved successful in many different applications.

## 2.2 Models to describe the gas explosion phenomena

The correct modelling of explosion involves the relevant parameters such as geometrical design and physical effects. Based on these characteristics, the available models to describe gas explosions behaviour are empirical and phenomenological models and numerical models using Computational Fluid Dynamics (CFD).



### 2.2.1 Empirical and Phenomenological models

The empirical and phenomenological models are concentrated on describing the major physical effects as much as possible, while the geometry is simplified. Such approaches require low computational effort and enable the analysis of different explosion scenarios in a short time.

Empirical models comprise the simplest class of models for gas explosion. They are based on correlations obtained by experimental results and includes venting guidelines, TNT equivalence model, TNO model, multi-energy and Congestion Assessment (CA) model.

The the TNT and TNO models are the most traditional empirical models for gas explosion analysis (WANG *et al.*, 2017) (KUHL *et al.*, 1994) (GUO *et al.*, 2016) (PITBLADO *et al.*, 2014). In TNT model the combustion energy is assumed to convert into an equivalent charge weight of TNT (2,4,6-trinitrotoluene) while TNO model assumes that all combustion energy present in the flammable part of a gas cloud contributes to explosion.

Even though, both TNT and TNO, as other empirical models, are easy to apply and give quickly initial responses, they are very conservative methods. They always assume that a detonation shock wave occurs when a gas explosion at the stoichiometry condition happens which gives low accurate results. Furthermore, using these empirical methods, some essential parameters must be selected by experience.

Phenomenological models are simplified physical models which attempt to describe the main physical process of an explosion based on idealized geometry and empirical correlation. The most representative tools for this class of models are CHICÉ and SCOPE (PARK; LEE, 2009). Phenomenological models give reasonable results with simple geometrical structures but are not useful for complex geometries.

### 2.2.2 CFD models

Computational Fluid Dynamic (CFD) models are based on the fundamental differential formulations that govern the explosion process. By using the Finite Volume Method, these equations are integrated over control volumes surrounding the relevant grid points in both space and time.

The numerical analysis using CFD is capable to account the geometrical details which are not possible when applying empirical or phenomenological models. The CFD solutions contain a great amount of information about the flow field (pressure, velocity, specific mass, etc...) which enables the understanding of its behaviour. However, the results must be analysed with care. Some errors can be generated during the numerical solution and the model will have low representation compared with the real problem (VERSTEEG;

MALALASEKERA, 2007).

### 2.2.2.1 CFD Codes for Gas Explosion

Due to the complexity of gas explosion, the CFD codes available for modelling such phenomena rely heavily on sub-models for the representation of small-scale objects, coupled with relatively simple numerical schemes for solution of the governing equations. Leake and Ledin (2002) point out some CFD codes for gas explosion analysis. Such codes are listed below:

- **EXSIM**

The EXSIM (EXplosion SIMulator) is a semi-implicit, finite volume code that uses a structured Cartesian grid. PDR (Porosity / Distributed Resistance) is used for modelling the small-scale geometries. The first and second order upwind schemes are applied to discretize the governing equations. The set of algebraic equations is solved using the TDMA (tri-diagonal matrix algorithm) method and the pressure correction method is used to correct the velocity components, pressure, and density, and guarantee mass conservation in the subsequent time step. Turbulence is modelled via the  $k - \varepsilon$  model and combustion is considered to take place in a single step mechanism (SAETER, 1998). The EXSIM code can be found into the *KFX*<sup>©</sup> CFD package, a ComputIT program (ComputIT).

- **FLACS**

FLACS (FLame ACceleration Simulator) code has been developed by CMR-GEXCOM. It is a finite volume code based on a structured Cartesian grid. The PDR approach is also used to model sub-grid scale obstacles. The turbulence is modelled by the  $k - \varepsilon$ . The discretization of the governing equations follows a upwind/central differencing scheme (second order accurate) and the combustion model treats the flame as a collection of flamelets (ARNTZEN, 1998).

- **AUTOREAGAS**

AutoReaGas is a three-dimensional finite volume code is also based on a structured Cartesian grid and that uses the PDR concepts to model small-scales geometries. The first order accurate Power Law scheme is used to discretize the governing equations and SIMPLE algorithm is implemented for pressure correction. The turbulence is modelled via the standard two equation  $k - \varepsilon$ . The combustion model assumes that the combustion reaction takes place as a single step process. Euler equations are solved for blast wave propagation (BERG *et al.*, 1994).

- **COBRA**

COBRA belongs to a class of advanced CFD explosion codes that uses unstructured grids and adaptive mesh algorithms. The code apply an explicit or implicit, second order accurate, finite volume integration scheme. The grid is unstructured and may be refined automatically locally within the flow. The PDR approach is used for modelling small-scale obstacles. Turbulence is modelled also using a  $k - \varepsilon$  turbulence model.

- **NEWT and MCNEWT-PDR**

NEWT is also an unstructured adaptive mesh, three dimensional, finite volume code. A second-order accurate discretization scheme is used for the convective fluxes and a fourth-stage Runge-Kutta time integration approach is used for the time dependent calculations. The NEWT code uses a variant of the  $k - \varepsilon$  turbulence model where the near wall damping function is dependent on the turbulence Reynolds number (WATTERSON *et al.*, 1998). The combustion is modelled using the eddy break-up model or a laminar flamelet model. The original NEWT solver was developed at the Engineering Department of the Cambridge University for turbo-machinery applications. The code has been improved over the years to include the PDR method to account for the resistance of small scales objects in gas explosion simulations (VIANNA; CANT, 2010).

Some of codes described above have been improved over the years while others have become obsolete. The available CFD codes for gas explosion analysis comprise commercial tools and most of them use the PDR concepts to assess the geometry. Even though the PDR modelling considerably reduces the computational costs in CFD simulation of gas explosion, there is no available literature concerning how to obtain the porosity values for a mesh calculation. In this work a new technique is presented to obtain a porous mesh for CFD calculations based on the Gilbert-Johnson-Keerthi distance algorithm.

## 3 RANS Numerical Modelling for Turbulent Premixed Combustion

In CFD, to solve any physical problem, the first step is to know well the phenomena and the system under study. It will allow the development of a mathematical model and to choose the best numerical tool. In mathematical modelling, it is important to include the identification of the system variables and their applications as well as the conditions of the study.

The equations that govern the flow are obtained applying the principle of conservation to the main quantities, such as mass, momentum and energy. Considering the Reynolds-Averaged Navier Stokes (RANS) approach, the equations that govern the flow are computed for mean quantities by averaging the instantaneous balance equations. This averaging procedure introduces unclosed quantities that have to be accounted by using additional models as well as for turbulence and combustion.

The modified governing equations and the dedicated developed models are described in this chapter. Particular attention is paid to the various source terms and the closure of the combustion reaction rate in the equation for the conservation of a scalar, namely progress variable.

This chapter presents the numerical formulation applied to handle the combustion process as gas explosion.

### 3.1 Mass and momentum equations

#### 3.1.1 The mass conservation equation

The mass conservation in the differential form is shown below. The equation is also known as the *Continuity equation*:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (3.1)$$

The first term in the left hand side represents the rate of change of fluid density with time. For flows in steady state this term is equal to zero. The second term is the divergence term. It indicates the rate of change of the specific mass with respect to the spacial coordinates.

### 3.1.2 The momentum conservation equation

The momentum equation, expressed by Equation 3.2, originates from the application of Newton's second law of motion to an element of fluid. It includes the principle of angular momentum and represents the variation of momentum as function of resultant forces acting on the fluid.

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \nabla \cdot (\rho \mathbf{u} \mathbf{u}) = \rho g - \nabla P + [\nabla \tau] \quad (3.2)$$

Equation 3.2 can be applied to all directions of fluid flow as long as an equation for the stress tensor is provided. Such equation that relates the stress tensor with the rate of fluid deformation is called a constitutive equation.

The first term in the left represents the rate of increase of momentum while the second takes into account the rate of momentum addition by convection and molecular transport.

The terms in the right represent the body force and pressure gradient, respectively. The last term in Equation 3.2 is a second order tensor that computes the viscous effects.

Similarly to Continuity Equation, the equation of motion can be simplified according to some flow features:

- The Navier-Stokes Equations applied to a Newtonian fluid and incompressible flow can be written as:

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \mathbf{u} \cdot (\nabla \rho \mathbf{u}) = \rho g - \nabla P + \mu \nabla^2 \mathbf{u} \quad (3.3)$$

- For an inviscid flow, the viscous effects could be negligible ( $\mu = 0$ ), giving the "*Euler Equation*":

$$\frac{\partial}{\partial t}(\rho \mathbf{u}) + \mathbf{u} \cdot (\nabla \rho \mathbf{u}) = \rho g - \nabla P \quad (3.4)$$

## 3.2 Elementary Description of turbulence

Because turbulence fluctuations increase the combustion reaction rate by enhancing the mixture between fuel and air, the turbulence plays an important role on gas explosion process.

Turbulence can be defined as an irregular motion of fluid flow in which the various quantities show a random variation with time and space coordinates, so that statistically distinct average values can be discerned (WILCOX *et al.*, 1993).

Kundu e Cohen (2002) mention some characteristics of turbulent flows:

- Irregularity - The flow is irregular, random and chaotic and consists a spectrum of different scales (eddy sizes). The largest eddies are of the order of the flow geometry and the smallest eddies are given by viscous forces.
- Large Reynolds Number - Turbulence occurs at high values of Reynolds number, when inertial forces become more important than the viscous forces.
- Effective transport - In turbulent flows, the fluid particles are fast mixed while they are moving through the flow. As consequence, the transport and fluid mixture is much more effective compared with laminar flow.
- Three dimensional - Turbulent flow is always three-dimensional. The motion of fluid particles occurs due to fluctuations in position and time in a three dimensional velocity field.
- Dissipation - Turbulent flow is dissipative, which means that kinetic energy in smallest eddies are transformed in internal energy. The kinetic energy is transferred through different sizes of eddies from the largest to the smallest. The process of energy transfer is called energy cascade process.
- Continuum - The turbulent flow can be treat as continuum, even though the smallest turbulent scales in the flow are larger than the molecular scale.

The available methods to solve the turbulence and calculate the quantities of interest are classified into two main groups.

In the first group, it can be found the models capable to completely solve mathematically the transport equations and, consequently, the turbulence. The main method used in these models are the *Direct Numerical Simulation* (DNS).

In DNS, the equations are directly solved for all length and time scales of fluid flow. This method can capture all relevant phenomena without approximation or simplification, including the smallest scales. DNS can be considered the best way to solve the turbulence, but because all length-scales and time-scales are solved, DNS is computationally expensive. As the computational cost increases with Reynolds number ( $\simeq Re^3$ ), DNS is restricted to flows with low to moderate  $Re$ .

Other turbulence models as *Reynolds Averaged Navier-Stokes* (RANS) and *Large Eddy Simulation* (LES) are not able to capture all phenomena that are involved at the the fluid flow. They are used to solve the transport equations for a mean velocity flow field and require less computational effort than DNS.

In RANS models all turbulent effects are modelled and the quantities are calculated based on their average values. Because RANS simulations do not directly solve the transport equations and turbulence scale, it is expected to be less accurate and reliable than LES and DNS. However, it has presented good performance for many engineering studies with low computational costs. RANS methods are the most widely used approach in CFD simulations of industrial flows.

The *Large Eddy Simulation* is a method that partly solve the transport equations. Andrey Kolmogorov has defined the small length scale as the ones that could exist in a fluid flow without energy dissipation (*Kolmogorov scale*). In LES, the equations are directly solved (as in DNS) for large turbulent eddies, with length down to the Kolmogorov scale. The effects in smallest eddies are modelled using a *sub-grid* model.

Concerning combustion process, DNS is applied for very simple systems to better understand and recognize the flame structure and flow regimes (as described in section 2.1.2.3). Both LES and RANS models are applicable for turbulent premixed combustion modelling. In principle, LES has the potential to provide more accurate and realistic results than RANS, but it is limited by the use of refined meshes and also computational costs. Thus most of turbulent premixed combustion modelling is conducted within the framework of traditional RANS approach.

### 3.3 RANS turbulence modelling

As consequence of the irregularity, the properties of a turbulent flow exhibit random fluctuations. Their behaviour can be defined as the sum of an average field and its fluctuations. Figure 3 shows an example of an instantaneous velocity and its fluctuations around the mean velocity in a turbulent flow.

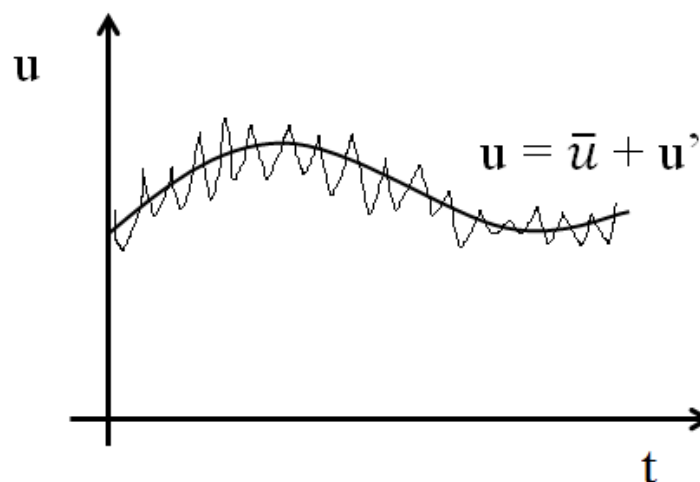


Figure 3 – Fluctuations in a turbulent flow - example of a velocity field.

Following the example shown in Figure 3, mathematically, the instantaneous value for any quantity can be expressed as the sum of the average values and their respective fluctuations:

$$\phi = \bar{\phi} + \phi' \quad (3.5)$$

where  $\phi$  is the value of the variable in a specific time,  $\bar{\phi}$  is its average time and  $\phi'$  is the turbulent fluctuation in an instant.

To illustrate the influence of turbulence fluctuations on the mean flow, the instantaneous equation for any quantity  $\phi$  of interest is considered:

$$\frac{\partial(\rho\phi)}{\partial t} + \nabla \cdot (\rho u\phi) = \nabla \cdot (\Gamma_\phi \nabla \phi) + S_\phi \quad (3.6)$$

where  $\Gamma_\phi$  is the diffusion coefficient and  $S_\phi$  is the source term.

Considering the Reynolds averaged approach, the quantity  $\phi$  can be expressed as a mean value  $\bar{\phi}$  and its fluctuating component  $\phi'$ , so  $\phi = \bar{\phi} + \phi'$ . Starting from an instantaneous balance (Equation 3.6):

$$\frac{\partial(\bar{\rho}\bar{\phi})}{\partial t} + \nabla \cdot (\bar{\rho}u\bar{\phi}) \rightarrow \frac{\partial(\bar{\rho}\bar{\phi})}{\partial t} + \nabla \cdot (\bar{\rho}\bar{u}\bar{\phi} + \overline{\rho'u'\phi'}) \quad (3.7)$$

Here, the unclosed quantity  $\overline{\rho'u'\phi'}$  represents the turbulent flux of the property  $\phi$  and need to be modelled.

Following the averaging procedure, the momentum equation (Equation 3.2) for the mean velocity component is modified:

$$\frac{\partial}{\partial t}(\bar{\rho}\bar{\mathbf{u}}) + \nabla \cdot (\bar{\rho}\bar{\mathbf{u}}\bar{\mathbf{u}}) = \bar{\rho}g - \nabla\bar{P} + [\nabla(\bar{\tau} - \overline{\rho'\mathbf{u}'\mathbf{u}'})] \quad (3.8)$$

In Equation 3.8,  $\tau$  is the Newton stress for the mean flow. The additional term due to the fluctuations,  $\overline{\rho'\mathbf{u}'\mathbf{u}'}$  is identified as an additional stress due to turbulence called *Reynolds Stress*. Therefore to handle this term turbulence models are applied.

The RANS turbulence models are divided into two main classes: the eddy viscosity models and Reynolds stresses models.

Eddy viscosity models consider that turbulence consists a small eddies which are continuously forming and dissipating. They also assume that Reynolds stresses are proportional to mean velocity gradients.



In Reynolds stresses turbulent models, all components of Reynolds stress tensor and the dissipation rate are presented in transport equations. Such equations are solved for the individual stress components. Here, the most common models are:

The correct choice of the most appropriate turbulence model is crucial to a successful modelling and CFD simulation of a real problem as turbulent and reactive flow.

### 3.3.1 Eddy Viscosity turbulence models

The eddy viscosity turbulence models are based on Boussinesq assumption (WILCOX *et al.*, 1993) which makes an analogy between the Newton stress and the Reynolds stress.

According to the Newton's law of viscosity, the viscous stress are proportional to the strain rate:

$$\tau = \mu \nabla \mathbf{u} \quad (3.9)$$

Following the same idea, according to the Boussinesq assumption, the Reynolds Stress are proportional to the mean strain rate:

$$\overline{\rho \mathbf{u}' \mathbf{u}'} = \mu_t \nabla \bar{\mathbf{u}} \quad (3.10)$$

where  $\mu_t$  is defined as turbulent or eddy viscosity.

The Boussinesq formulation needs a relationship to express the Reynolds tensor from the mean velocity gradient, and other quantities as the turbulent kinetic energy  $k$ , to achieve the closure of the mean equations. Therefore, a linear constitutive equation was defined:

$$\mathbf{R} = -2\nu_t \mathbf{S} \quad (3.11)$$

where  $\nu_t$  is the turbulent kinematic viscosity;  $\mathbf{R}$  is the anisotropic tensor, and  $\mathbf{S}$  is the strain rate tensor.

The strain rate tensor  $\mathbf{S}$  computes the mean velocity gradient:

$$\mathbf{S} = \frac{1}{2} \nabla \bar{\mathbf{u}} \quad (3.12)$$

The anisotropic tensor  $R$  accounts for the turbulent kinetic energy  $k$ . Considering Cartesian co-ordinates so that the velocity vector  $\mathbf{u}$  has x-component  $u$ , y-component  $v$  and z-component  $w$ , the value of  $k$  can be obtained through:

$$k = \frac{1}{2}(u^2 + v^2 + w^2) \quad (3.13)$$

so,

$$R = \overline{\mathbf{u}'\mathbf{u}'} - \frac{2}{3}kI \quad (3.14)$$

here,  $I$  is the unit tensor.

The main Eddy Viscosity turbulence models are:

- *Zero-Equation* or algebraic model;
- *One-Equation* models;
- *Two-Equation* models:  $k$ - $\varepsilon$ , RNG  $k$ - $\varepsilon$  and  $k$ - $\omega$ .

As the name suggests, the algebraic model describe the stresses by means of simple algebraic formulae for  $\mu_t$  as a function of the spatial position, while the One-Equation models solve one turbulent transport equation, usually the turbulent kinetic energy.

The two-equation turbulence models comprises a complete turbulence approach based on the Boussinesq assumption. They have been widely used to simulate the turbulent flows in different engineering applications. These models have two independent transport equations, one for turbulent kinetic energy, and the other for turbulent dissipation rate (for the  $k$ - $\varepsilon$  model) or specific dissipation rate (for the  $k$ - $\omega$  model).

A briefly explanation of the  $k$ - $\omega$  model is presented below. Dedicated section, with more details, is given to the  $k$ - $\varepsilon$  model, once it is adopted in this work.

### 3.3.1.1 The $k$ - $\omega$ turbulence model

The standard  $k$ - $\omega$  turbulence model was introduced by Wilcox (1988). The model formulation is given by:

$$\mu_t = \frac{\rho k}{\omega} \quad (3.15)$$

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho \mathbf{u} k) = \nabla \cdot \left( \frac{\mu_t}{\sigma_k} \nabla k \right) + 2\mu_t E_{i,j} E_{i,j} - C_\mu \rho k \omega \quad (3.16)$$

$$\frac{\partial}{\partial t}(\rho \omega) + \nabla \cdot (\rho \mathbf{u} \omega) = \nabla \cdot \left( \frac{\mu_t}{\sigma_\omega} \nabla \omega \right) + 2\mu_t E_{i,j} E_{i,j} \frac{\alpha \rho}{\mu_t} - \beta \rho \omega^2 \quad (3.17)$$

where  $\sigma_k$ ,  $C_\mu$ ,  $\alpha$ , and  $\beta$  are the model constants.

The main advantage of the  $k - \omega$  model is the near wall treatment. However, the model has a potential to overprediction of eddy viscosity in large normal-strain flow regions. The model has been improved during the years to account for a better approach away from the surface (MENTER, 1994).

### 3.3.2 The $k-\varepsilon$ turbulence model

The  $k-\varepsilon$  model is considered the industry standard model and has proven to be stable and numerically robust. For general purpose simulations, this model offers a good compromise of accuracy and robustness. The main drawbacks in  $k-\varepsilon$  turbulence model includes low accuracy for unconfined flows and weak shear layers (for wakes and mixing layers) (VERSTEEG; MALALASEKERA, 2007).

The  $k-\varepsilon$  turbulence model was developed based on the mechanisms that affect the turbulent kinetic energy (LAUNDER; SPALDING, 1983). The two independent transport equations applied in this model computes the turbulent kinetic energy  $k$  and the turbulence dissipation rate  $\varepsilon$ . The terms  $k$  and  $\varepsilon$  are used to define the velocity scale  $\vartheta$  and length scale  $l$  for large scale turbulence representation:

$$\vartheta = k^{1/2} \quad (3.18)$$

$$l = \frac{k^{3/2}}{\varepsilon} \quad (3.19)$$

defined

Based on the velocity scale  $\vartheta$  and length scale  $l$  defined by Equations 3.18 and 3.19, the turbulent eddy viscosity can be specified:

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon} \quad (3.20)$$

where  $C_\mu$  is a constant of the model.

The values of  $k$  and  $\varepsilon$  can be obtained from differential transport equations as follows:

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho \mathbf{u} k) = \nabla \cdot \left( \frac{\mu_t}{\sigma_k} \nabla k \right) + 2\mu_t E_{i,j} E_{i,j} - \rho \varepsilon \quad (3.21)$$

$$\frac{\partial}{\partial t}(\rho \varepsilon) + \nabla \cdot (\rho \mathbf{u} \varepsilon) = \nabla \cdot \left( \frac{\mu_t}{\sigma_\varepsilon} \nabla \varepsilon \right) + \frac{\varepsilon}{k} (C_{1\varepsilon} 2\mu_t E_{i,j} E_{i,j} - C_{2\varepsilon} \rho \varepsilon) \quad (3.22)$$

where  $C_{1\varepsilon}$ ,  $C_{2\varepsilon}$ ,  $\sigma_k$  and  $\sigma_\varepsilon$  are constants.  $\mu_t E_{i,j} E_{i,j}$  represent the turbulence production due to viscous forces and  $\rho \varepsilon$  is the viscous dissipation.

Considering Equations 3.20 - 3.22, it is observed that the  $k$ - $\varepsilon$  model presents five adjustable parameters. For the standard model, they assume the values listed in Table 1.

Table 1 – Constant values for the standard  $k$ - $\varepsilon$  turbulence model (LAUNDER; SPALDING, 1983).

Constant	Value
$C_\mu$	0.09
$C_{1\varepsilon}$	1.44
$C_{2\varepsilon}$	1.92
$\sigma_\varepsilon$	1.00
$\sigma_k$	1.30

In the current research, the wrinkling of the flame area is due to the turbulent velocity fluctuations. The turbulence model is coupled to the rate of reaction via Equation 3.48. The fluctuating part of the velocity is calculated using the kinetic turbulent energy  $k$ .

### 3.3.2.1 Turbulence close to walls

Due to the presence of the boundary layer near the walls,  $k$  and  $\varepsilon$  assumes a specific behaviour in this region. The approach to describe the quantities depends on the Reynolds number.

At high Reynolds number, the standard  $k$  -  $\varepsilon$  model uses the "log-law" velocity behaviour and considers that the rate of turbulence production equals the rate of dissipation. Therefore wall functions are defined:

$$u^+ = \frac{U}{u_\tau} = \frac{1}{\kappa} \ln(Ey_p^+) \quad (3.23)$$

$$k = \frac{u_\tau^2}{\sqrt{C_\mu}} \quad (3.24)$$

$$\varepsilon = \frac{u_\tau^3}{\kappa y} \quad (3.25)$$

where  $\kappa$  is the Von Karman's constant, taken as 0.41; and  $E$  is the wall roughness parameter  $E = 9.8$ .

At low Reynolds number, the log-law is not valid and  $k - \varepsilon$  model must be modified (VERSTEEG; MALALASEKERA, 2007) to couple the effect of low Reynolds number into the model equations. Therefore, Equations 3.20 to 3.22 become:

$$\mu_t = C_\mu \rho f_\mu \frac{k^2}{\varepsilon} \quad (3.26)$$

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho \mathbf{u} k) = \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right) + 2\mu_t E_{i,j} E_{i,j} - \rho \varepsilon \quad (3.27)$$

$$\frac{\partial}{\partial t}(\rho \varepsilon) + \nabla \cdot (\rho \mathbf{u} \varepsilon) = \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right) + \frac{\varepsilon}{k} (C_{1\varepsilon} f_1 2\mu_t E_{i,j} E_{i,j} - C_{2\varepsilon} f_2 \rho \varepsilon) \quad (3.28)$$

Here,  $f_\mu$ ,  $f_1$  and  $f_2$  are wall damping functions modelled according to Lam e Bremhorst (1981):

$$f_\mu = 1 - \exp(-0.0165 Re_t^2) \left( 1 + \frac{20.5}{Re_t} \right) \quad (3.29)$$

$$f_1 = \left( 1 + \frac{0.05}{f_\mu} \right)^3 \quad (3.30)$$

$$f_2 = 1 - \exp(-Re_t^2) \quad (3.31)$$

The same line of reasoning applied to low Reynolds flows at regions near the wall is suggested in the current work for the initial phase of burning in the reacting flow.

As it will be discussed next, the initial phase of the kernel formation of the reaction zone is modelled balancing the contribution from laminar flow and turbulent flow up to a pre-established ratio between inertial effects and viscous forces. Equation 3.53 allows for a smooth transition where the damp function  $f_\mu$  has been inspired in the formulation discussed above.

### 3.4 Favre averaging

When using Reynolds Averaged approach, the unclosed quantity  $\overline{\rho' u' \phi'}$  present in the left hand side of Equation 3.7 can act as a source term of  $\phi$  for the mean flow field and consequently the average quantity may not be conserved in a steady flow (POINSOT; VEYNANTE, 2005). Specially for flows with significant density variation, such a condition is difficult and also awkward to handle in CFD. To solve this problem, the concept of Favre averages can be used instead Reynolds averages.

The Favre averages comprise mass-weighted averages (FAVRE, 1969), so each quantity  $\phi$  is density-weighted before averaging:

$$\tilde{\phi} = \frac{\overline{\rho\phi}}{\bar{\rho}} \quad (3.32)$$

and the fluctuation from Favre average is given as:  $\phi'' = \phi - \tilde{\phi}$

The main advantage of using Favre averages is that the terms contain correlations with density fluctuations, thus leading to a simplification of the averaged equations (CANT; MASTORAKOS, 2008).

Considering the Favre average concept, the mean balance equation for mass conservation becomes:

$$\frac{\partial \bar{\rho}}{\partial t} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}}) = 0 \quad (3.33)$$

Starting from the complete Navier-Stokes equation (Eq. 3.2), the averaged balance equation for momentum conservation is given as:

$$\frac{\partial}{\partial t}(\bar{\rho} \tilde{\mathbf{u}}) + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{\mathbf{u}}) = \bar{\rho} g - \nabla \bar{P} + [\nabla(\bar{\tau} - \bar{\rho} \widetilde{\mathbf{u}'' \mathbf{u}''})] \quad (3.34)$$

### 3.5 Numerical Combustion Modeling

As previously discussed in Section 2.1.2, for most explosion simulations the combustion process is modelled as a single step reaction. The Flamelet model stands out in this approach by considering that reactions take place in a thin and wrinkled surface separating the unburned reactants from the fully burned products.

### 3.5.1 Turbulent premixed combustion modelling by Bray-Moss-Libby

Considering the classical flamelet approach developed by Bray *et al.* (1985), a reaction progress variable "c" is assigned zero value in the unburned reactants region and a value of unity in the fully burned products stabilisation zone. Thus:

$$c = \frac{(Y_F - Y_{FR})}{(Y_{FP} - Y_{FR})} \quad (3.35)$$

where  $Y_F$  is the fuel mass fraction and the subscripts R and P refer to reactants and products, respectively.

The transport equation for the Favre-average reaction progress variable is given as:

$$\frac{\partial}{\partial t} \bar{\rho} \tilde{c} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{c}) = \nabla \cdot \left( \left( \frac{\mu + \mu_t}{\sigma_c} \right) \nabla \tilde{c} \right) + \bar{\omega} \quad (3.36)$$

In Equation 3.36 above, the summ  $\mu + \mu_t$  denotes the effective viscosity and  $\sigma_c$  is the turbulent Schmidt number for the reaction progress variable. The last term  $\bar{\omega}$  describes the mean reaction rate source term.

Similarly, for the mixture fraction (air and fuel), the transport equation for the mixture fraction  $\tilde{F}$  is expressed as:

$$\frac{\partial}{\partial t} \bar{\rho} \tilde{F} + \nabla \cdot (\bar{\rho} \tilde{\mathbf{u}} \tilde{F}) = \nabla \cdot \left( \left( \frac{\mu + \mu_t}{\sigma_f} \right) \nabla \tilde{F} \right) \quad (3.37)$$

Here,  $\sigma_f$  is the turbulent Schmidt number for the mixture fraction. Both  $\sigma_c$  and  $\sigma_f$  are assumed to have a value equal to 0.7.

#### 3.5.1.1 Thermodynamics and Equation of State

Following the BML formulation, the main thermodynamic variables must be coupled to the reaction progress variable. Because in explosion phenomena the Mach number may not be always considered to be low, such link can not be done under assumptions of constant enthalpy. Hence, an additional transport equation must be included to take into account the total energy  $\tilde{E}$  (CANT; BRAY, 1989):

$$\frac{\partial}{\partial t} \bar{\rho} \tilde{E} + \nabla \cdot \bar{\rho} \mathbf{u} \left( \tilde{E} + \frac{\bar{P}}{\bar{\rho}} \right) = \nabla \cdot \left( \left( \frac{\mu + \mu_t}{\sigma_E} \right) \nabla \left( \tilde{E} + \frac{\bar{P}}{\bar{\rho}} \right) \right) + \nabla \cdot \mathbf{u} \bar{\tau} \quad (3.38)$$

where the total energy  $\tilde{E}$  is defined as:

$$\tilde{E} = \tilde{e} + \frac{1}{2}\tilde{u}^2 + \tilde{k} \quad (3.39)$$

in which  $e$  is the internal energy, so that:

$$\tilde{e} = \int_{T_0}^{\tilde{T}} C_v dT + H(Y_{FR}(1 - \tilde{c}) + Y_{FP}\tilde{c}) \quad (3.40)$$

where  $T$  is the temperature,  $C_v$  is the constant volume specific heat capacity and  $H$  is the energy of combustion.

The pressure  $\bar{P}$  is obtained from a equation of state of ideal gas, valid for small changes in molar mass:

$$\bar{P} = \frac{\bar{\rho}R\tilde{T}}{\bar{W}_M} \quad (3.41)$$

Here,  $R$  is the universal gas constant,  $\tilde{T}$  is the Frave mean temperature and  $\bar{W}_M$  is the mean molar mass. The Frave mean temperature and the mean molar mass can be obtained by Equations 3.42 and 3.43 bellow, respectively:

$$\tilde{T} = \frac{\tilde{e}}{C_{VP}} + T_R(1 - \tilde{c}) \left(1 - \frac{C_{VR}}{C_{VP}}\right) - \frac{H}{C_{VP}} [(1 - \tilde{c})Y_{FR} + \tilde{c}Y_{FP}] \quad (3.42)$$

$$\bar{W}_M = \frac{W_R(1 - \tilde{c}) + W_P\tilde{c}(1 + \bar{\tau})}{1 + \bar{\tau}\tilde{c}} \quad (3.43)$$

In Equations 3.38 and 3.43  $\tau$  is the heat release parameter, defined as:

$$\bar{\tau} = \left(\frac{\rho_R}{\bar{\rho}_P}\right) - 1 \quad (3.44)$$

### 3.5.1.2 Reaction rate modelling

To close the set of equations in the turbulent premixed combustion modelling it is necessary to define the reaction rate source term  $\bar{\omega}$  present in Equation 3.36.

The Bray-Moss-Libby flamelet model requires the specification of a probability density function to model the reaction rate. The formulation considers that during the turbulent reaction phase, the mean reaction rate in a probe location is defined as follow:

$$\bar{\omega} = \rho_R u_L \sum \quad (3.45)$$



Here,  $\rho_R$  is the density of the reactants,  $u_L$  is the laminar flame speed and  $\Sigma$  is the flame surface area per unit of volume. This last variable is obtained by considering that the passage of laminar flamelets in a specific point is a stochastic process (CANT; BRAY, 1989):

$$\Sigma = \frac{g\tilde{c}(1 - \tilde{c})}{|\hat{\sigma}_y|\hat{L}_y} \quad (3.46)$$

In Equation 3.46 above,  $g$  is a constant and  $|\hat{\sigma}_y|$  the orientation factor, both evaluated from experimental data, therefore  $g \simeq 1.5$  and  $|\hat{\sigma}_y| \simeq 0.5$ .  $\hat{L}_y$  is the integral length scale of wrinkling and is linked with the laminar flamelet length scale:

$$\hat{L}_y = c_L l_L f\left(\frac{u'}{u_L}\right) \quad (3.47)$$

where the constant  $c_L \simeq 1.0$  and  $l_L = (\nu/u_L)$ .

The function  $f$  in Equation 3.47 is obtained based on experimental observations. The factor  $\Sigma$  increases with  $u'/u_L$  up to a maximum value and tends to decrease when the reaction rate is reduced by flame stretch (CANT; BRAY, 1989), so:

$$f\left(\frac{u'}{u_L}\right) = \left[ \frac{1}{1 + \frac{C_{w1}}{(u'/u_L)}} \left( 1 - \exp\left[ \frac{-1}{1 + C_{w2}(u'/u_L)} \right] \right) \right]^{-1} \quad (3.48)$$

The function  $f$  is used to obtain the flame length scale, based on the fact that such parameter is defined in response to perturbations from the turbulent velocity field. Such function is calibrated from experimental data and the values of constants  $C_{w1}$  and  $C_{w2}$  are taken as 1.5 and 4.0, respectively.

The laminar flame speed  $u_L$  is obtained by empirical correlation (ABU-ORF, 1996), which eliminates the need of flamelet libraries:

$$u_L = a\Phi^b \exp[-c(\Phi - d)^2] \left(\frac{T_R}{T_0}\right)^\alpha \left(\frac{P}{P_0}\right)^\beta \quad (3.49)$$

Here,  $\Phi$  is the equivalence ratio (between the mass of fuel and air within the flammability limits) and  $a$ ,  $b$ ,  $c$ ,  $d$ ,  $\alpha$  and  $\beta$  are fuel dependent constants. Table 2 presents the values of such constants for methane and propane.

### 3.5.2 Initial phase laminar combustion

As addressed in Chapter 2, the overall behaviour of combustion phenomena during an explosion is based on the turbulent mechanisms. However, it is important to

Table 2 – Constant values for the laminar flame speed correlation (ABU-ORF, 1996).

fuel	a	b	c	d	$\alpha$	$\beta$
<b>Methane</b>	0.6097	-2.554	7.3105	1.2303	2.0	-0.42
<b>Propane</b>	0.4243	0.7345	4.5003	0.9813	1.77	-0.25

understand that an initial burn phase is present at the beginning of the process. Once ignition has occurred, the flame develops initially at the laminar speed. As the kernel grows, the laminar flames instabilities cause the flame to wrinkled and accelerate until the turbulent combustion is set. Therefore, to model the explosion phenomena, a special treatment must be given to this initial phase.

Birkby *et al.* (2000) propose a numerical approach to represent the initial laminar flame behaviour. The flame burns at laminar speed by scaling the sum of the reaction rates in each node and it is equal to the total reaction rate of a laminar flame with the same total surface area. Therefore, Equation 3.45 becomes:

$$\bar{\omega}_{lam} = \rho R u_L A_{flame} \quad (3.50)$$

The flame surface area ( $A_{flame}$ ) is obtained on a geometrical basis for spherical flames, until the flame reaches an obstacle and distorted. This geometrical relation to obtain the flame area for a given volume of a sphere (ARNTZEN, 1998) is given by:

$$A_{flame} = \pi^{1/3} (6V)^{2/3} \quad (3.51)$$

In Equation 3.51 the area of the flame is calculated by identifying all cells contain the progress variable  $\tilde{c} = 0.5$  contour.

The reaction rate for each node is calculated in the same way for the standard turbulent combustion model, since the proportion to the total laminar burning rate is calculated for each node. In this case, the source term in equation 3.36 is given as:

$$\bar{\omega} = \rho R u_L C A_{flame} \frac{\tilde{c}(1 - \tilde{c})}{\sum_{nodes} \tilde{c}(1 - \tilde{c})V} \quad (3.52)$$

where C is a fuel dependent constant.

Another method has been proposed by Vianna e Cant (2014) in order for the transition from laminar to turbulent combustion. The model assumes that such transition

occurs when the turbulent Reynolds number is equal to a predefined value ( $Re \approx 500$ ). By using this model, the reaction rate source term (equation 3.36) is calculated as follows:

$$\bar{\omega} = f_{\mu}\omega_{turb} + (1 - f_{\mu})\omega_{lam} \quad (3.53)$$

where  $\omega_{turb}$  is the original source term of BML model given in equation 3.45 and  $\omega_{lam}$  is the laminar source term proposed by Arntzen (1998) (Equation 3.52).

In this case the area of the flame is given as:

$$A_{flame} = \pi^{4/9}(V)^{2/3} \quad (3.54)$$

and  $f_{\mu} = \max(0; 1 - \exp(-0,008(Re_{turb} - Re_{500})))$ , as previously discussed.

## 3.6 The Porosity Distributed Resistance Method (PDR)

The concept of Porosity Distributed Resistance (PDR) was introduced by Patankar e Spalding (1974) to handle multi obstacles flows. Initially, the method was implemented in heat exchanger geometries and was progressively applied for complex geometries.

The main idea of PDR is to consider the small scales objects that are not solved by the computational mesh as a porous media. This medium would offer a resistance to the fluid flow (SAVILL; SOLBERG, 1994). This technique allows the use of a much coarser computational mesh than the one required to resolve the fine scales of the geometry, which in turn permits the use of a less detailed specification of the geometry.

The PDR model has been used in several different models to assess gas explosion in complex geometries (HJERTAGER *et al.*, 1992), (VIANNA; CANT, 2010), (SAVILL; SOLBERG, 1994) and (FOTHERGILL *et al.*, 2003).

### 3.6.1 PDR Formulation

In PDR formulation, the presence of small scales objects modifies the governing equations. Due to the presence of obstructions in the computational cell, only part of the total volume is available to the flow. Moreover, the small objects offer additional resistance to flow and also turbulence production (HJERTAGER *et al.*, 1992).

Considering a typical control volume, shown in Figure 4, the porosity or volume fraction occupied by the fluid can be defined as:

$$\beta_v = \frac{V_f}{V_f + V_s} = \frac{V_f}{\Delta x \Delta y \Delta z} \quad (3.55)$$

where  $\beta_v$  is the porosity volume,  $V_f$  is the volume of fluid and  $V_s$  is the volume of a solid geometry.

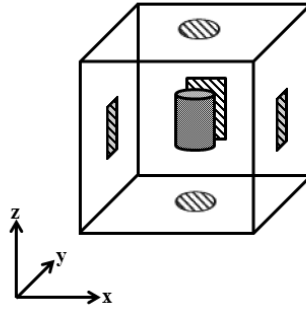


Figure 4 – Typical computational cell partially occupied by a solid object and the projected area in the cell faces.

The volume porosity is calculated based on the volume of solid not resolved by the mesh and the volume of the computational cell itself. In a similar manner an area porosity is calculated using the projected area of the the obstacle on the cell face. The area porosity is defined in each of the three directions (x, y and z). The expression for x direction is given as:

$$\beta_x = \frac{\int_{surface} dydz}{\Delta y \Delta z} \quad (3.56)$$

Similar expressions can be obtained for porosities in y and z directions. All the volume/area porosities will have values ranging from 0 (completely blocked) to 1 (completely open). Such porosity is taken into account by the governing equations to calculate the fluid flow properties only in the region of the computational cell that is available to the flow.

### 3.6.2 Modified Conservation Equations by Using the Porosity Distributed Resistance (PDR) Model

According to the porosity distributed resistance method, porosity values are assigned to faces and volume of each computational cell and must be considered in each

conservation equation. Thus, the terms of equations that contain area and volume are amended by a new variable, namely  $\beta$ .

Furthermore, the presence of obstacles that are not solved by the mesh cause an additional resistance to the flow and also the production of turbulence that must be accounted for.

### 3.6.2.1 Additional resistance to the flow

The resistance that unresolved obstacles offer to the mean flow is modelled as a frictional resistance (SHA; LAUNDER, 1979):

$$R_j = -f_j A_w \frac{1}{2} \rho U_j^2 \quad (3.57)$$

Here  $A_w$  is the wetted area per unit volume:  $A_w = A_j(\beta^{-1} - 1.0)^2$ . The friction factor  $f_j$  depends on different parameters such as velocity, porosity value, characteristic length or hydraulic diameter, shape and orientation.

Patankar e Spalding (1974) present two different models to calculate the friction factor based on the flow orientation to an obstacle. For flows parallel to an obstacle  $f_j$  is calculated through Equation 3.58 while for normal flows  $f_j$  is given by Equation 3.59

$$f_j = 0.048 Re_D^{-0.2} \quad (3.58)$$

$$f_j = 0.23 + \frac{0.11}{\left[ \left( 3 \frac{P}{D} \right)^{0.5} - 1 \right]^{1.08}} Re_D^{-0.15} \quad (3.59)$$

In both equations,  $Re_D$  is the Reynolds number based on the hydraulic diameter. In Equation 3.59 the term  $D$  represents the diameter of an obstacle and  $P$  the space between the obstacles.

As consequence of this additional resistance to the flow, the momentum equation is modified. Hence, Equation 3.2 becomes:

$$\frac{\partial}{\partial t}(\beta_v \rho \mathbf{u}) + \mathbf{u} \cdot \nabla(\beta_i \rho \mathbf{u}) = \beta_v \rho g - \beta_v \nabla P + [\beta_i \nabla \cdot \boldsymbol{\tau}] + \beta_i R \quad (3.60)$$

### 3.6.2.2 Turbulence production due to resistance

The presence of a porous media also represents an additional source of turbulence fluctuations. Sha e Launder (1979) have proposed two models in order to describe this additional turbulence production.

The first model takes into account the turbulence production due to the presence of shear layers on the sides of the unresolved obstacles. The modified formulation, suggested by Vianna e Cant (2010) and given by:

$$S_{k1} = C_S \mu_t \beta_v [(U - U_S)^2 + (V - V_S)^2 + (W - W_S)^2] A_w^2 \quad (3.61)$$

where  $U_S$ ,  $V_S$  and  $W_S$  represents the slip velocities and are taken to be a fraction of the values at cell faces was used in the current research.

In the second model, the turbulence production due to the wakes of the unresolved obstacles is calculated:

$$S_{k2} = C_B \rho U_j^3 A_w \quad (3.62)$$

The constants  $C_S$  and  $C_B$  present in Equations 3.61 and 3.62 are adjustable parameters ( $C_B = C_S/2$ ).

The additional production terms described here  $S_{k1}$  and  $S_{k2}$  are introduced in the  $k - \varepsilon$  turbulence model by increasing the turbulence production term. Equations 3.27 and 3.28 become:

$$\frac{\partial}{\partial t}(\rho k) + \nabla \cdot (\rho \mathbf{u} k) = \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_k} \right) \nabla k \right) + S_T - \rho \varepsilon \quad (3.63)$$

$$\frac{\partial}{\partial t}(\rho \varepsilon) + \nabla \cdot (\rho \mathbf{u} \varepsilon) = \nabla \cdot \left( \left( \mu + \frac{\mu_t}{\sigma_\varepsilon} \right) \nabla \varepsilon \right) + \frac{\varepsilon}{k} (C_{1\varepsilon} f_1 S_T - C_{2\varepsilon} f_2 \rho \varepsilon) \quad (3.64)$$

where  $S_T$  represents the source of turbulence production and is given by

$$S_T = 2\mu_t E_{i,j} E_{i,j} + S_{k1} + S_{k2}$$

## 4 Porosity Flow Solver (PFS)

In gas explosion, as well as in almost all real problems, the solution of the differential equations that represent the fluid dynamics is not straightforward, even when using numerical techniques.

Specifically for the momentum conservation, the presence of the non-linear terms in the Navier-Stokes equations greatly increases the complexity of the numerical solution.

The strategy adopted in this work relied on a bi-dimensional solver where the Euler equation is used to solve the momentum conservation as basis for the development of the three-dimensional Navier-Stokes. The initial code, called *Euler Solver*, is customised to solve compressible flows in sonic and supersonic flow regimes (GONÇALVES; VIANNA, 2014).

The *Euler Solver* was improved step by step until the complete three-dimensional Navier-Stokes solver for gas explosion in complex geometries was engineered. For the sake of clarity, the 3D Navier-Stokes is named PFS (Porosity Flow Solver) and this abbreviation will be used along the remaining of this thesis.

The four main steps considered were:

- First, a three-dimensional Euler code was developed based on the existing 2D *Euler Solver*. Since there is no need for calculation of the viscous term, the implementation of the Euler solver is easier than the implementation of the complete Navier-Stokes equations. It can also serve as the framework for future development of the detonation module of the code.
- In the second part, the PDR (Porosity Distributed Resistance) was included in the code to represent the geometry under study.
- Step three embedded the development of the three-dimensional Navier-Stokes code. The viscous effects were included into the momentum equation so that the complete Navier-Stokes equations were solved. In this step, the turbulence model was also implemented;
- In the fourth and last part, the combustion modelling was coupled with the porosity concept and was coded into the framework of the 3D Navier-Stokes program. The flamelet concept was used at this step.

As result of the process listed above, the three-dimensional CFD code was developed. The main features of the code and details of the architecture are discussed in this chapter. More information about the code routines and the link between them are found in Appendix .

## 4.1 Structure of PFS

The CFD codes generally present four main elements: the geometry generator, the pre-processor (mesh and set up), the solver, and the post-processor. This section outlines the various aspects of the development of the computational tool.

### 4.1.1 Geometry

Although the utilisation of PDR techniques are focused on the small scales of a geometry that is not resolved by the computational mesh, the current research investigates how the full geometry can be parametrised using the collision algorithm for convex sets.

The initial step of the simulation using PFS comprises the conversion of the geometry in a porous media. The area and volume porosities are calculated as described in Section 3.6. The porosity calculation is performed by the supporting code, namely **PrePro PFS**. The porosity code was developed to check the collision between two convex sets. In this particular case, the computational mesh was assumed to be a convex set and the Cartesian coordinates of the geometrical model was model as a second set (MOREIRA *et al.*, 2015). Stereolithography CAD files in "stl" format can be *read* in **PrePro PFS** that uses the Gilbert-Johnson-Keerthi Distance algorithm (GJK algorithm) to parametrise the geometry as porous media.

The outputs of this process are data files containing values for the areas and volumes porosities. It also provides the parameters used to build the porous mesh (dimensions of the computational domain and size of the cells) (MOREIRA *et al.*, 2015).

#### 4.1.1.1 Fundamentals of the GJK Algorithm

The Gilbert-Johnson-Keerthi Distance algorithm (GJK algorithm) is a method for obtaining the minimum distance between two convex objects based on the concept of the Minkowski difference ((GILBERT; FOO, 1989), (GILBERT *et al.*, 1988)). The algorithm can also be used to check for collision between two convex objects.

Given two sets  $A, B$  in  $R^n$  of position vectors in the Euclidean space, the Minkowski addition is formed by adding every vector in A to every vector in B (Equation



4.1).

$$A + B = \{a + b \mid a \in A, b \in B\} \quad (4.1)$$

Similarly, the Minkowski difference is formed by subtracting every vector in B from every vector in A (Equation 4.2).

$$A - B = \{a - b \mid a \in A, b \in B\} \quad (4.2)$$

In Minkowski difference, if the two sets are colliding, the resulting difference will contain the zero vector.

The main advantages of the GJK algorithm is the computational velocity and stability. There are many techniques that make possible to further increase the efficiency of the algorithm (ERICSON, 2004), (BERGEN, 1999). The algorithm also relies on a support function that makes possible to check for collisions between any kind of discrete and analytical objects, in any kind of format.

One important idea of the GJK algorithm is to try to interactively build a polytope, polygon (2D) or polyhedron (3D), on the Minkowski difference that encapsulates the origin. This polygon or polyhedron can be defined as a simplex. The simplex is build using a support function, that returns a point on shape that is furthest in a direction  $D$ . Using the furthest point in one direction is important because it creates the simplex with the largest area (or volume) and increases the chance of the algorithm to converge quickly. The furthest point in a direction of the Minkowski difference of two sets of vectors  $P$  and  $Q$  is the maximum of the dot product of the direction and the Minkowski difference (Equation 4.3).

$$\begin{aligned} \text{support} &= \max((P - Q) \cdot D) \\ &= \{\max((p - q) \cdot D) \mid p \in P, q \in Q\} \\ &= \{\max(p \cdot D - q \cdot D) \mid p \in P, q \in Q\} \\ &= \{\max(p \cdot D) - \max(q \cdot (-D)) \mid p \in P, q \in Q\} \\ &= \text{support}(P, D) - \text{support}(Q, -D) \end{aligned} \quad (4.3)$$

To initialise the algorithm, the supported function is set up using a random direction and its result is added to the simplex. After that, the negative vector is used as initial direction vector for the main loop. The main loop of the function performs three actions. First it adds a new vector  $A$  to the simplex by calling the support function using the direction vector  $D$ . Then, it checks if this new vector  $A$  is in the same direction as the direction vector  $D$ . If it is not, then the algorithm tries to get to a point as far as

possible in the direction  $D$  but could not get pass the origin. This means that there are no points past the origin, otherwise the simplex will never be able to build a polytope that encapsulates the origin. If the check return true, then the point  $A$  is added to the the simplex  $S$  and the function Nearest is called.

The last step of the main loop is the Nearest function. This function first checks if the origin is inside the polytope. If this is true, then the function returns true and the algorithm ends. If this is false, it may removes old points of the simplex  $S$  that are no longer required and a new direction  $D$  to search for. This function is build based on the dimension (2D or 3D) of the problem. Algorithm 4.1 is a resumed explanation of the GJK algorithm.

---

**Algorithm 4.1** GJK Algorithm
 

---

**Require:**  $P$  = list of points of solid 1

**Require:**  $Q$  = list of points of solid 2

**Require:**  $D_0$  = initial search direction

**function** GJK( $P, Q, D_0$ )

**Initialize:**  $A \leftarrow \text{Support}(p, D_0) - \text{Support}(q, -D_0)$

**Initialize:**  $S \leftarrow \{A\}$

$\triangleright S$  is the simplex list

**Initialize:**  $V \leftarrow -A$

**loop**

$A = \text{Support}(p, D) - \text{Support}(q, -D)$

**if**  $\text{dot}(A, D) < 0$  **then**

**return** False

**end if**

$S \leftarrow A$

$S, D, col := \text{Nearest}(S)$

**if**  $col$  **then**

**return** True

**end if**

**end loop**

**end function**

**function** SUPPORT( $X, D$ )

**return**  $\max(X, D)$

**end function**

---

#### 4.1.1.2 Porosity Mesh Calculation

The parametrised geometry can be obtained using the GJK algorithm to check for a collision between a mesh element and each geometric object. This process is described in Figure 5 and is conducted in three main steps. In the first step a geometric object is placed into a domain; on step 2 a structured mesh is created over the domain; and in the last step 3 the GJK technique is used to calculate the porosity of the mesh. In this approach, every element fully occupied by a solid has a value of 0.0, while every empty element has a value of 1.0. If an element of the mesh is half full the value of 0.5 is assigned

for the local porosity of the computational cell.

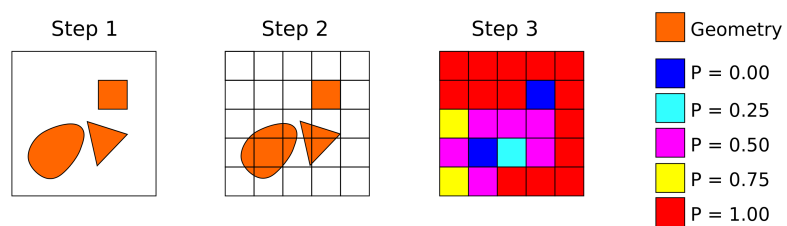


Figure 5 – Steps of porosity mesh calculation.

### • Porosity Volume

The porosity volume is obtained applying the 3D GJK algorithm and the procedure described above. As the method uses a structured mesh, each computational element will be a hexahedron (cube). Algorithm 4.2 below describes the process.

---

#### Algorithm 4.2 Volume Porosity Algorithm

---

**Require:**  $S$  = list of solids in the geometry

**Require:**  $N$  = list of elements of the mesh

**Require:**  $P$  = list of porosity values

**Initialize:**  $P \leftarrow 1$

**for all**  $s \in S$  **do**

**for all**  $n \in N$  **do**

$P(n) := P(n) - \text{collision3D}(n, s)$  ▷ collision returns 1.0 or 0.0

**end for**

**end for**

---

### • Porosity Area

The values of porosity area are obtained by "slicing" each geometric solid on the required plane. A trivial slicing method, described by Gregori Neri Volpato e Silva (2014) is used. As a structured mesh is used, the values of porosity area are calculated on the faces of the hexahedrons. Algorithm 4.3 describes the process for calculation of porosity area over x direction.

---

#### Algorithm 4.3 Area Porosity Algorithm (x direction)

---

**Require:**  $S$  = list of solids in the geometry

**Require:**  $N_x$  = list of elements of the mesh in the x direction

**Require:**  $P_x$  = list of area porosity values in the x direction

**Initialize:**  $P_x \leftarrow 1$

**for all**  $s \in S$  **do**

**for all**  $n_x \in N_x$  **do**

$s_x := \text{slice}(s, n_x)$

      ▷ slice the solid  $s$  at the  $(x, 0, 0)$  coordinate of  $n$  on the  $x$  direction

      ▷ slice the solid  $s$  at the  $(x, 0, 0)$  coordinate of  $n_x$  on the  $x$  direction

$P_x(n) := P_x(n) - \text{collision2D}(n_x, s_x, 1)$  ▷ collision returns 1.0 or 0.0

**end for**

**end for**

---

### • Recursion

Because the GJK algorithm can only return true or false, a recursive division approach is applied to obtain refined porosity values. An example of this procedure is presented in Figure 6 where two mesh elements are shown near a solid obstacle. In na

first step the GJK algorithm identifies that the solid object is in contact with the bottom element but is not in contact with the top element, so the top element is immediately assigned a porosity value of 1.0 while recursion starts at the bottom element. On the next step, the bottom element is divided into  $k$  sub-elements and each sub-element is checked with the GJK algorithm again. To obtain the final porosity value each sub-element is subtracted from the main element according to its value. If it is not in contact, the value is zero. If it is in contact, the value is  $\frac{1}{k^n}$ .

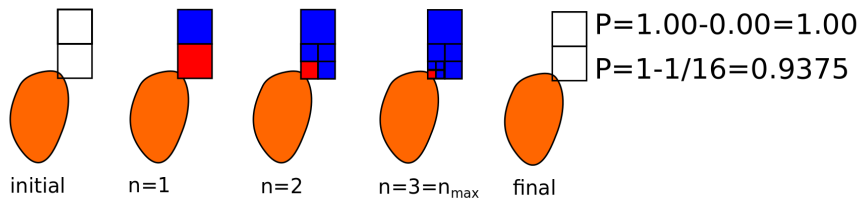


Figure 6 – Porosity recursion.

The refined porosity volumes are obtained breaking every mesh element (cube) in 8 equal elements, while the refined porosity areas are obtained dividing each face of the mesh (square) into 4 equal faces. The new collision function is described by algorithm 4.4, with  $k = 8$  for the 3D problem and  $k = 4$  for the 2D problem.

---

**Algorithm 4.4** Collision Recursion
 

---

```

1: function COLLISION( $n, s, int$ )                                ▷ int is the number of the interaction
2:    $col :=$  GJK( $n, s$ )                                           ▷ GJK return true or false for collision
3:   if  $col = true$  then
4:     if  $int \leq int_{max}$  then
5:        $M :=$  divide( $n, k$ )                                       ▷ divide  $n$  into  $k$  sub-elements
6:                                             ▷  $M =$  list of sub-elements
7:        $sum := 0$ 
8:       for all  $m \in M$  do
9:          $sum := sum + collision(m, s, int + 1)$ 
10:      end for
11:      return  $sum/k$ 
12:     else
13:       return 1
14:     end if
15:   else
16:     return 0
17:   end if
18: end function

```

---

- Results for Porosity mesh calculation

Figure 7 shows the developed porosity method applied to a simplified oil platform STL model. The model is composed of 26 convex objects (cylinders, spheres and hexahedrons), mesh size is  $100 \times 100 \times 64$  (617463 cells). Using a FORTRAN OMP code, with 8 cores, the simulation took around 10 seconds to complete (MOREIRA *et al.*, 2015).

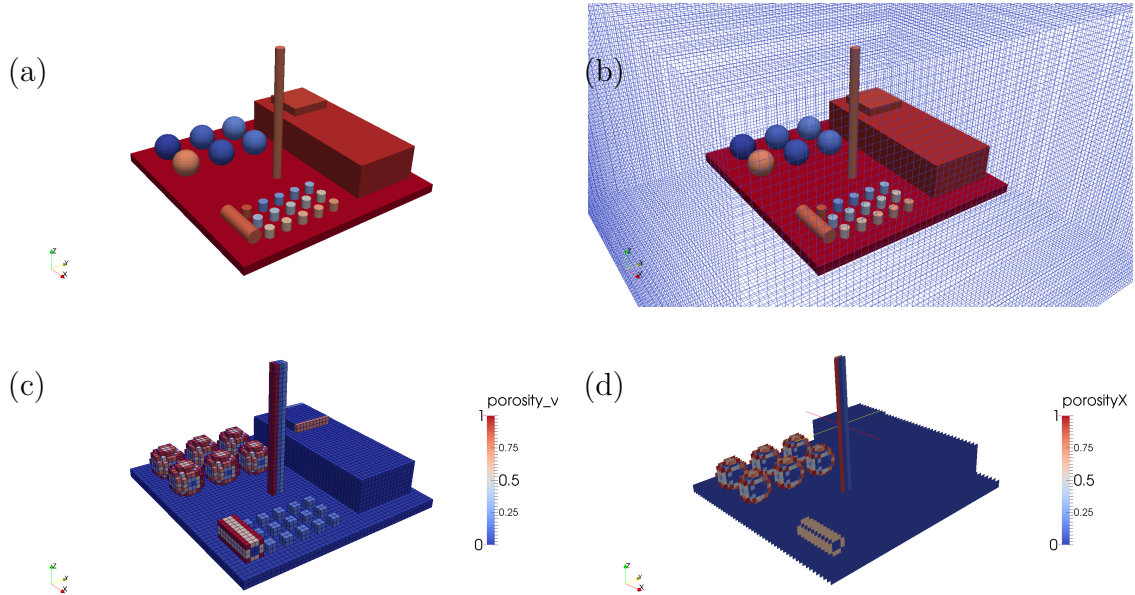


Figure 7 – Porosity applied to a simple oil platform model: (a) the STL geometry; (b) Platform geometry and mesh; (c) Platform porosity volume; (d) Platform porosity area (in X direction). Results are post-processed using Paraview.

### 4.1.2 Preprocessing

In PFS code, all the partial differential equations are integrated over a control volume, following the Finite Volume Method (FVM).

The Finite Volume Method is a spatial discretization technique for partial differential equations. By using this method, the approximate equations are obtained by integration of the governing equation in its conservative form. This discretization process gives rise to a linear system of algebraic equations for the variables that can be numerically solved (MALISKA, 2004). The FVM is in common use for discretizing computational fluid dynamics equations once the resulting of the process is similar to perform a balance of the property over a discrete control volume.

#### 4.1.2.1 Variable Storage

In finite volume method the control volumes can be defined in two different ways: cell-centred and cell-vertex. In the cell centred approach the element mesh is used as a control volume and the computational nodes are considered as the centre of this control

volume, where the unknowns are stored. In the cell vertex approach the unknowns are defined at the corners of the cell and the control volume is defined by all computational cells surrounding a specific node. Figure 8 presents an example of a mesh with these two different approaches.

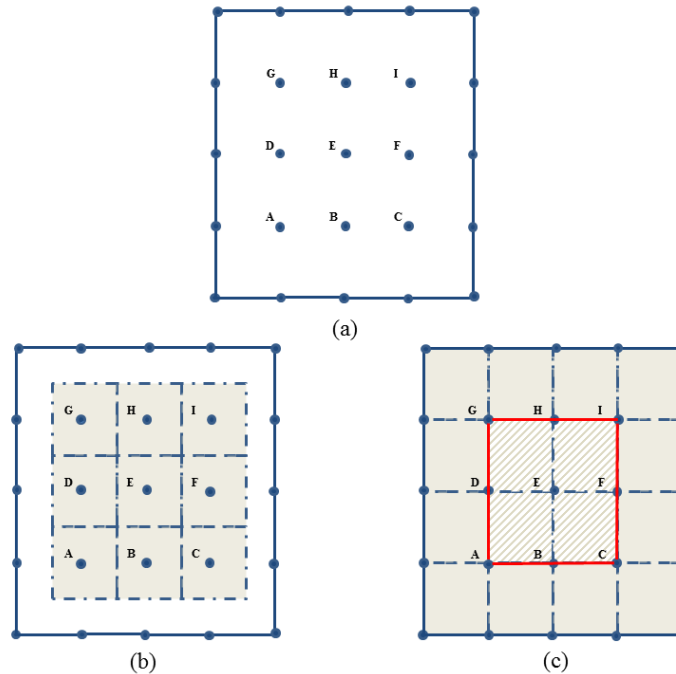


Figure 8 – General grid in finite volume method: nodes (a); cell-centred control volumes (b) and cell-vertex control volume (c).

The cell-centred approach is an obvious choice for control volumes, once they coincide with the mesh cells. However the cell-vertex methodology presents more flexibility for the definition of control volume. The advantages of cell-vertex approach include more accuracy; less CPU time and memory requirement and easy treatment of boundary conditions (DISKIN *et al.*, 2010) (HEJRANFAR; AZAMPOUR, 2015). In PFS, the cell-vertex approach was considered to represent the control volumes.

#### 4.1.2.2 PFS Mesh

Due to the complexity of the phenomenon, structured grids have been used in almost of all CFD codes for gas explosion ((LEA; LEDIN, 2002), (WATTERSON *et al.*, 1998), (HJERTAGER *et al.*, 1992) ). Hirsch (2007) emphasizes that structured grids, compared with unstructured grids, are often more efficient from CFD point of view, in terms of accuracy, CPU time and memory requirement. In this work the structured grid is adopted.

In PFS code, the mesh is built on a Cartesian coordinate system with X, Y, Z (i,j,k) being the right-handed set. The overall strategy of mesh generation is illustrated

in Figure 9. The final mesh is structured and composed by discrete cells defined by a set of vertices, faces, and edges as shown in Figure 10 (vertices - (a), faces - (b) and elements - (c)).

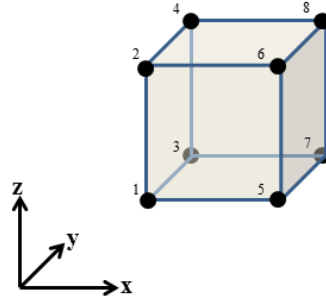


Figure 9 – The hexahedral computational cell in PFS code.

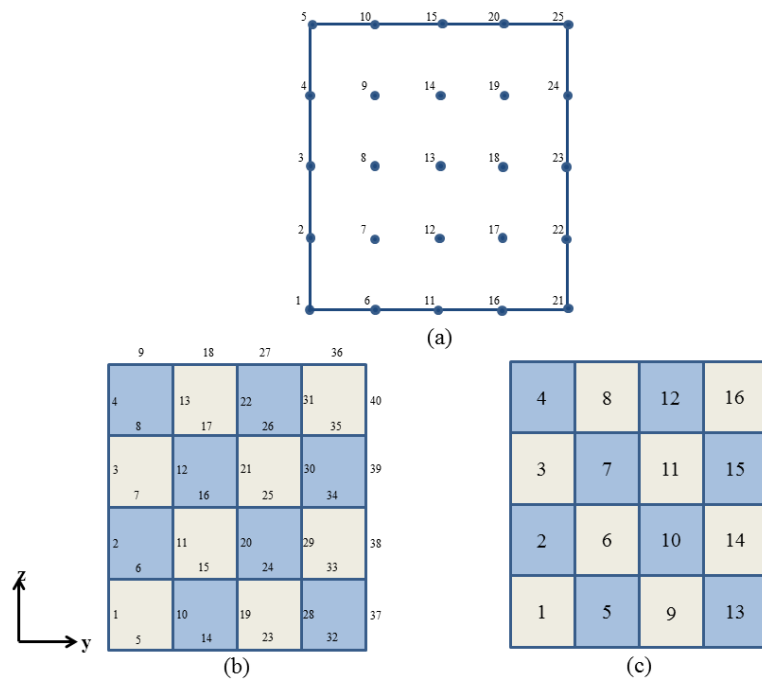


Figure 10 – Two-Dimensional Mesh Discretization: nodes (a); faces (b) and elements (c).

The geometric information about each cell is stored into a matrix that is used to make the elements connectivity (as described in Moukalled F.; Mangani e Darwish (2016)). It is possible to identify the neighbour cells and vertex and the shared and boundary faces. In Figure 10 (c), for example, the element 7 is composed by the vertex: 8, 9, 14 and 13; by the faces: 16, 12, 17, 21; and has the elements 3, 8, 6 and 11 as neighbours.

The area of each cell face and the volume of each computational cell are evaluated according to Hirsch (2007). Following this strategy, the area of a quadrilateral element is calculated as a half of the vector cross products of the diagonals. The area



vector attached to a cell face is derived from the divergence theorem which indicates that the sum over all surface normals of any closed cell must be zero. Such condition is assessed when calculating the cell face areas. Another important consideration is that all areas must come out positive as a way to ensure the flow through the computational cell.

To obtain the volume of the computational cell, simple formulae are used for basic cells such as tetrahedra, pyramids or hexahedral. For the calculation of the volume of an hexahedron, its volume can be divided into three pyramids with a specific point as summit. The volume of each pyramid is described as a third of the product between the area of the pyramid base and the vector orthogonal to the associated surface. The hexahedral volume is obtained by the sum of the volume of the three pyramids. As example, in Figure 11, the hexahedron 1-2-3-4-5-6-7-8 is divided into three pyramids with point 4 as summit.

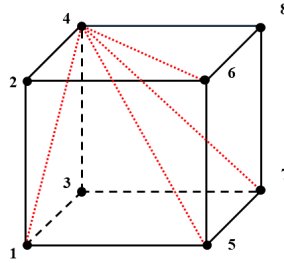


Figure 11 – Subdivision of an hexahedral volume in pyramids.

The volume of the hexahedron given above is calculated as:

$$Vol_{HEXA} = Vol_{1573-4} + Vol_{5687-4} + Vol_{1265-4} \quad (4.4)$$

### 4.1.3 PFS Solver

The solution technique adopted in PFS is the *Lax-Friedrichs* method described in Hirsch (2007). This method consists in a central difference scheme that is modified to stabilise the unstable forward in time marching approach.

The traditional second order central discretisation in space and first order explicit forward difference in time, for one dimension, is defined as:

$$\phi_i^{n+1} = \phi_i^n - \frac{\alpha}{2}(\phi_{i+1}^n - \phi_{i-1}^n) \quad (4.5)$$

where  $\phi$  represents the value of the variable of interest,  $i$  represents the point position in space and  $n$  is the time level. The parameter  $\alpha$  represents the *Courant Number* (CFL), the unique parameter of the scheme.

In Equation 4.5, if the term  $\phi_i$  in the right hand side is replaced by  $(\phi_{i-1} + \phi_{i+1})/2$ , an error of the order of  $\Delta x$  is introduced, reducing this scheme to first order in space, so:

$$\phi_i^{n+1} = \frac{1}{2}(\phi_{i+1}^n + \phi_{i-1}^n) - \frac{\alpha}{2}(\phi_{i+1}^n - \phi_{i-1}^n) \quad (4.6)$$

Equation 4.6 above describes the *Lax-Friedrichs* method, which is first order accuracy in space and time. Because it is explicit in time, this scheme is computationally efficient and simple to implement. However the scheme tends to give solutions containing non-physical oscillations or wiggles in regions where high gradients are experienced. The solution might be stabilised by adding artificial viscosity (smoothing).

#### 4.1.4 Postprocessing

All the postprocessing in PFS (the grid and the results visualization) is conducted in the open source tool *Kitware Paraview*.

The mains steps of a simulation using the developed code are presented in Figure 12. Following the CFD modelling, the blocks in gray represent the pre-processing step, the blue blocks represent the solver and the red one is used to describe the post-processing.

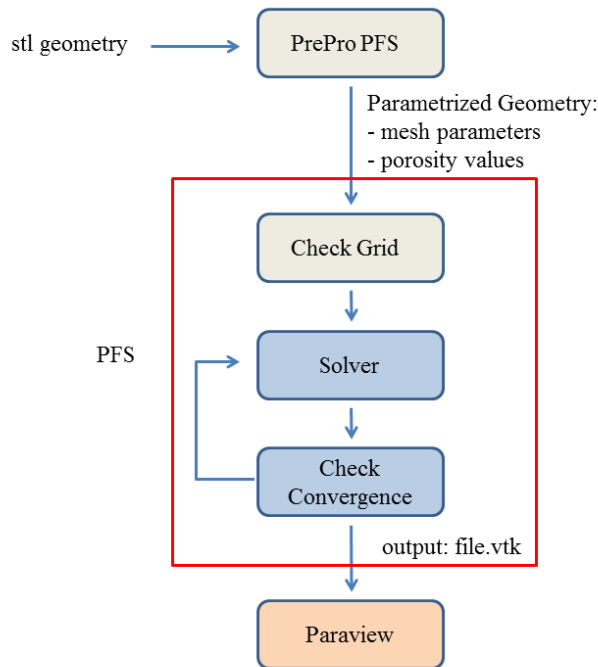


Figure 12 – Main steps of a simulation using PFS code.

It is possible to observe that the output of the **PrePro PFS** (files contain mesh parameters and porosity values for a specific geometry) is the input of the first subroutine

of PFS called "*check grid*". In this subroutine, the computational grid is rebuild and porosity values are attached for each computational cell.

## 4.2 Discretisation Process

During the PFS discretisation process, the variables are stored at the vertex cells (as described in Section 4.1.2). Figure 13 shows a typical set of hexahedral cells surrounding and influencing a node.

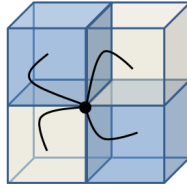


Figure 13 – Set of hexahedral computational cells surrounding and influencing a node in PFS code.

For better visualisation, Figure 13 shows only four cells. However it is possible to understand that in three dimensions a specific node is influenced by eight cells (four in each plane).

### 4.2.1 Discretisation of the Conservation Equations

The conservation equations of mass, momentum, energy and reaction progress variable, described in Chapter 3, have a similar format. Using a generic variable  $\phi$ , all these quantities can be expressed as:

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho u_j \phi)}{\partial x_j} = \frac{\partial}{\partial x_j} (\Gamma_\phi \frac{\partial \phi}{\partial x_j}) + S_\phi \quad (4.7)$$

The integration of the Equation 4.7 over a control volume V gives:

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV = - \int_V \frac{\partial(\rho u_j \phi)}{\partial x_j} dV + \int_V \frac{\partial}{\partial x_j} (\Gamma_\phi \frac{\partial \phi}{\partial x_j}) dV + \int_V S_\phi dV \quad (4.8)$$

In Equation 4.8, the convection and diffusion terms can be transformed into surface integrals by using the Gauss Divergence Theorem. The integral form of the equation is written as:

$$\int_V \frac{\partial(\rho\phi)}{\partial t} dV = - \int_S (\rho u_j \phi) dS_j + \int_S \Gamma_\phi \frac{\partial \phi}{\partial x_j} dS_j + \int_V S_\phi dV \quad (4.9)$$

The terms in Equation 4.9 represents the unsteady on the left hand side and the convection, diffusion and source terms on the right hand side, respectively.

As previously described, when the PDR method is applied, the transport equations are modified to take into account the porosity values of volume and areas for each computational cell. Hence, Equation 4.9 above becomes:

$$\int_V \frac{\partial(\rho\phi)}{\partial t} \beta_V dV = - \int_S (\rho u_j \phi) \beta_j dS_j + \int_S \Gamma_\phi \frac{\partial\phi}{\partial x_j} \beta_j dS_j + \int_V S_\phi \beta_V dV \quad (4.10)$$

where  $\beta$  is the assigned cell porosity.

#### 4.2.1.1 The Unsteady Term

The unsteady term in the left hand side of the Equation 4.10 is discretised considering the value at node shared by the eight computational cells surrounding it. The central approximation is used as described in Section 4.1.3.

#### 4.2.1.2 The Convection Term

For the convection term, the variables are assumed to have a piece-wise linear variation over the cell faces between the vertices. The flux sum for a given cell is computed as:

$$\frac{\partial(\rho u_i \phi)}{\partial x_j} = \sum_{nf=1}^6 (\rho u_i \phi A_i) \quad (4.11)$$

#### 4.2.1.3 The Diffusion Term

The derivative terms in the viscous stresses are piece-wise constant over the cell and are computed by simple application of the Gauss divergence theorem:

$$\left( \frac{\partial\phi}{\partial x_j} \right) = \frac{1}{VOL} \sum_{nf=1}^6 \phi \Delta A_{xj} \quad (4.12)$$

where  $nf$  indicates the face number of the control volume.

In Equation 4.12, the sum over the volume include the six (6) faces that belongs the hexahedral computational cell.

Using all the cells surrounding a specific node as control volume, the viscous stress terms at such node can be evaluated:

$$\frac{\partial}{\partial x_j} \left( \mu \left( \frac{\partial \phi}{\partial x_j} \right) \right) = \frac{1}{VOL} \sum_{nc=1}^8 \mu \left( \frac{\partial \phi}{\partial x_j} \right) \Delta A_{xj} \quad (4.13)$$

Here,  $nc$  represents the number of cells surrounding a node. Once the domain is discretised in hexahedral cells with cell vertex storage variable, in three dimensions, each node will receive the contribution of the eight surrounding cells (as visualized in Figure 13).

#### 4.2.1.4 The Source Term

The source term is different for each variable in individual conservation equation and it depends on what is being transported.

Numerically, for the momentum conservation, the source term represents the effect of pressure gradient. In case of the energy equation, it includes contributions due to pressure work, viscous dissipation and chemical source term.

#### 4.2.1.5 The Complete Discretised Conservation Equation

Considering the discretisation of each term of the Equation 4.10, the discretised generic equation solved in PFS is:

$$\boxed{\frac{\Delta(\rho\phi\beta_V V)}{\Delta t} = - \sum_{nf=1}^6 (\rho u_i \phi \beta_i A_i) + \sum_{nf=1}^6 \left( \rho \Gamma_\phi \beta_i A_i \frac{\partial \phi}{\partial x_j} \right) + S_\phi \beta_V V} \quad (4.14)$$

### 4.2.2 The time Advancement

The discretised equations are time-marched using a fourth order Runge-Kutta algorithm with residual smoothing as suggested by Jameson e Baker (1987). The variable of interest is updated by:

$$\begin{aligned} f_\phi^0 &= f_\phi^n \\ f_\phi^1 &= f_\phi^0 + \alpha_1 \Delta t f_{\phi v}^0 \\ f_\phi^2 &= f_\phi^0 + \alpha_2 \Delta t f_{\phi v}^1 \\ f_\phi^3 &= f_\phi^0 + \alpha_3 \Delta t f_{\phi v}^2 \\ f_\phi^4 &= f_\phi^0 + \alpha_4 \Delta t f_{\phi v}^3 \\ f_\phi^{n+1} &= f_\phi^4 \end{aligned}$$

The values of  $f_\phi^n$  and  $f_\phi^n + 1$  corresponding to the values at the beginning and the end of the  $n^{th}$  time step, while  $f_{\phi v}$  represents the net fluxes of the variable in a time step. The coefficients of the scheme are:  $\alpha_1 = 1/4$ ;  $\alpha_2 = 1/3$ ;  $\alpha_3 = 1/2$  and  $\alpha_4 = 1$ .

#### 4.2.2.1 Time Step

The PFS code runs in the unsteady form so a constant time step is applied. The time step is calculated based on the size of smallest computational cell and its value can be modified in accordance with the *Courant* number.

$$\Delta t = CFL \times \frac{\Delta x}{2 \times ss} \quad (4.15)$$

Here,  $\Delta t$  is the time step,  $\Delta x$  is the smallest size of the computational grid, and  $ss$  is the local sound speed.  $ss$  is calculated based on the properties of the fluid (calorific capacity and specific heats).

#### 4.2.2.2 Iterative Convergence

As PFS runs in unsteady form, for steady problems, the code solve the unsteady form of the governing equations and "*march*" the solution in time until the solution converges to a steady solution.

In steady state, the value of a specific variable  $\phi^n$  tends to be similar to the value calculated at the previous time step,  $\phi^{n-1}$ , and the errors tend to zero. Hence, the iteration process continues until the measure of the difference between  $\phi^{n-1}$  and  $\phi^n$  is "small enough ". Such difference known as residual is calculated in each grid point for each variable of interest as:

$$Residual = \frac{\sqrt{\sum_{i=1}^N (\phi_i^n - \phi_i^{n-1})^2}}{\sum_{i=1}^N \phi_i^n} \quad (4.16)$$

#### 4.2.3 Artificial Viscosity/Dissipation

The concept of artificial dissipation or artificial viscosity is introduced in numerical simulations to remove the oscillations around discontinuities in central differencing schemes. This additional term simulates the effects of the physical viscosity by introducing an error that is proportional to the second order derivatives of the variables.

In PFS, the artificial viscosity is applied to the variables after each time-step. The additional term is obtained by:  $SF \times \Delta^2 \phi$ , where  $SF$  represents a smooth factor that is proportional to the time step and the CFL number.

## 4.3 Boundary Conditions

- **Inflow**

At the inlet, the total pressure, temperature, turbulent kinetic energy and dissipation and velocity are specified. The derivative of the static pressure in the stream-wise is set to zero.

- **Outflow**

At the outflow boundary the static pressure is specified and the other variables are extrapolated from interior points.

- **Solid Surfaces**

Solid surfaces are set to have zero normal fluxes of mass, momentum and energy. The turbulent kinetic energy and the normal gradient of dissipation rate are also imposed to be zero on the surfaces.

- **Porosity Regions**

At the porosity regions the variables are modelled as solid surfaces (with zero normal gradient). Specifically for turbulence, the equations of  $k - \varepsilon$  model are modelled considering the approach of *low Reynolds number* described in Section 3.3.2.1.

### 4.3.1 Initial Condition

For the initial condition values of static pressure (P), density ( $\rho$ ) and temperature (T) are set in the computational domain. The pressure and temperature are set as reference values equal to  $10^5$  Pa and 300 K, respectively, while the gas density is set according to the fuel that is used in the simulation. The velocity field is obtained according to a specified pressure gradient.

#### 4.3.1.1 Ignition

The ignition starts by initialising a predefined ignition region (or point) where the fraction of the products is set (the progressive variable is equal to 1). The presence of products increases the temperature and decreases the density and hence a flow is induced in front of the flame and the combustion process is initialised. The initial flame speed and combustion process is handled by the laminar combustion model as explained in Section 3.5.2.

## 4.4 Set up and solver parameters

The "set up" comprises the pre-processor step where both, the physical and mathematical modelling are specified for the simulation. In the developed code, the "set up" is introduced by a text file that contains some thermodynamics properties of the fluid, flow conditions and solver parameters. Table 3 displays these main inputs.

Table 3 – Main set up features for PFS simulations.

<b>Parameter</b>	<b>Condition</b>
Fluid Type	0 or 1: Set zero to set air or 1 to set other fluid.
$C_p$ , $\gamma$ and $\mu$	Fluid properties
$P_{ref}$ and $T_{ref}$	Reference Pressure and temperature, normally set as 100 kPa and 300K, respectively.
$P_{down}$	Pressure specified at the outlet boundary.
<i>Velocity</i>	Prescribed velocity specified at the inlet boundary
CFL, SF and Residual	Values of CFL number, Smooth Factor and convergence criteria are specified here.
NSTEP and DUMP	Maximum number of iterations and number of iterations to dump a result
NS	0 or 1: Set 1 to solve the viscous term.
COMB	0 or 1: Set 1 to solve a combustion flow.



## 5 Case Studies for non-reacting flows

As described in Chapter 4, the Porosity Flow Solver has emerged from the bi-dimensional Euler solver and it was improved step by step until becoming a three-dimensional Navier-Stokes code customised for gas explosion modelling. It turns out, however, that fluid mechanics must be correctly modelled to be able to handle reacting flows.

This Chapter presents the results obtained for various flows in the absence of chemical reactions.

The simulations were performed using the 3D Euler solver considering both a standard geometry and a parametrised geometry as porous media. The chapter also comprises detailed analysis of the main parameters concerning the discretised equations. Particular attention is given to the smooth factor that is responsible for smoothing out the wiggles generated by the central difference scheme.

Prior to starting the comparison with experimental data for cold flows, numerical findings for a flow over bluff body were compared with simulations conducted by ANSYS-CFX. The point here was to verify if the implementation of the finite volume method and the associated models were properly implemented.

In the second part of the tests, simulations of turbulent flows were conducted for different parametrised geometry by solving the Navier-Stokes equations coupled with the  $k - \varepsilon$  turbulence model and the results were compared with experimental data.

### 5.1 Verification Procedure

#### 5.1.1 3D Euler simulations

Since the initial 2D Euler solver was tested for different conditions of compressible flow and high Mach number and came out with satisfactory results (GONÇALVES; VIANNA, 2014), the same tests were performed with the 3D solver.

The 3D duct geometry with a stricture in the centre was considered bounded by upper and lower solid walls. The size of the channel was set as  $10\text{m} \times 3.0\text{m} \times 3.0\text{m}$  in x, y and z directions, respectively, and it was divided into computational cells with 0.15m of length. Initially, a subsonic air flow with 35 m/s was set from left to right in x direction. Figure 14 shows the computational mesh and the simulation results.

Analysis of Figure 14 (b) shows that the Mach number (ratio between the local velocity and the local sound speed) is less than one, which corresponds to the simulation

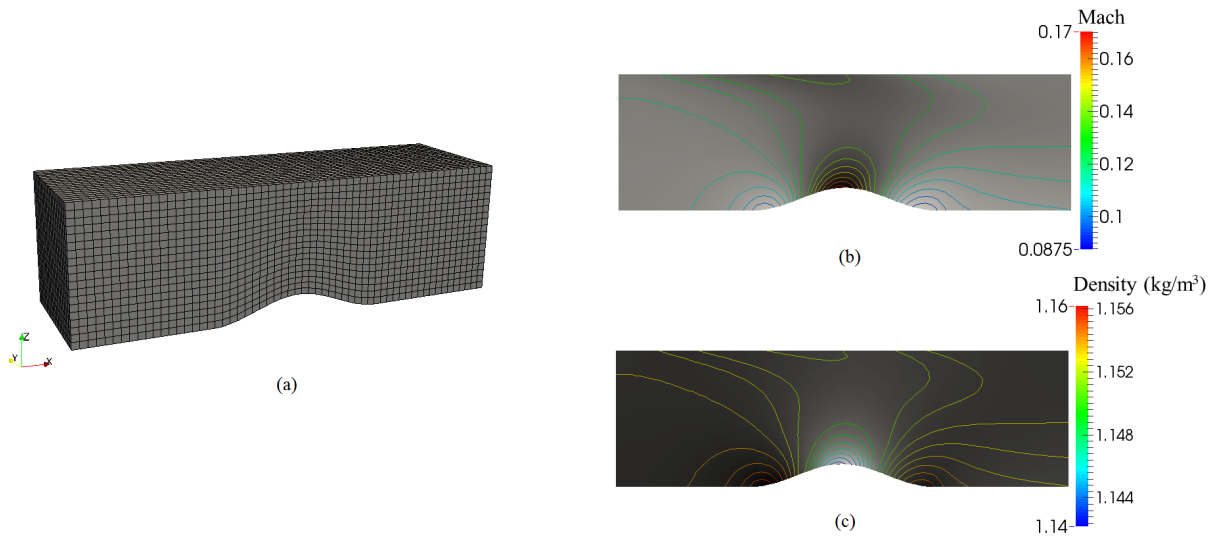


Figure 14 – Simulation of a subsonic flow in a three dimensional channel with a stricture in the center by solving Euler equation. Computational domain divided into hexahedral cells (a). (b) values of Mach number (b) and density field (c).

of subsonic flow. By analysing the Mach number values, it is also verified that the velocity increases at the bump region where the channel area available to the flow decreases. A slight variation in the density field is observed (Figure 14 (c)), which indicates minor compressibility effects when low velocities are applied.

An additional test considered supersonic flow. For gas flow, the compressible effects become relevant when the flow speed is high enough to allow the Mach number increase above 0.3.

Air flow with velocity of 280 m/s was assigned at the same lefthand boundary as shown in Figure 14 (a). Figure 15 shows the results of such simulation. In this case, a shockwave is observed downstream the channel strangulation (Figure 15 (a) ). Shockwaves are characterised by abrupt changes in flow conditions thus giving rise to discontinuity regions. Such regions appear under sonic and supersonic conditions. Furthermore significant changes in the density field is also verified (Figure 15 (b) ).

### 5.1.2 Verification of the 3D Euler code with a parametrised geometry

To analyse how the geometry parametrised by the GJK algorithm could be coupled with the 3D Euler solver and used in fluid flow simulations, a flow around a bluff body (a squared cylinder) was considered. At this stage of the development of PFS code, the porosity values associated with each computational cell was modelled in the set of conservation equations of mass, momentum (Euler) and energy. The additional resistance to the flow described in Section 3.6.2.1 was also included in the momentum equation.

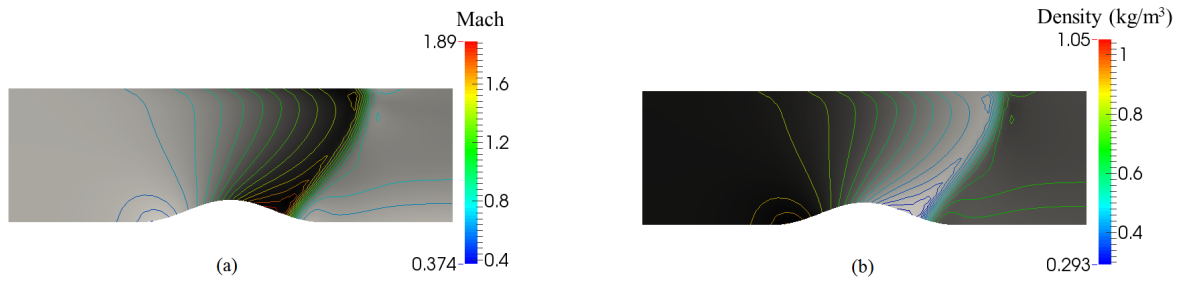


Figure 15 – Simulation of a supersonic flow in a three dimensional channel with a stricture in the center by solving Euler equation (a) values of Mach number (b) and density field (c) for an incompressible flow.

A "box" .stl geometry was created in the open source "Blender" CAD tool measuring 1.0m of length in each direction (x, y and z). The **PrePro PFS** tool was used to built the computational domain and to calculate the porosity values. The domain comprised a rectangular box with 12 m in x direction and 3m in y and z directions. The cube was placed in the computational domain and computational cells (0.15m) were considered. Figure 16 shows the .stl CAD geometry and its corresponding parametrised version.

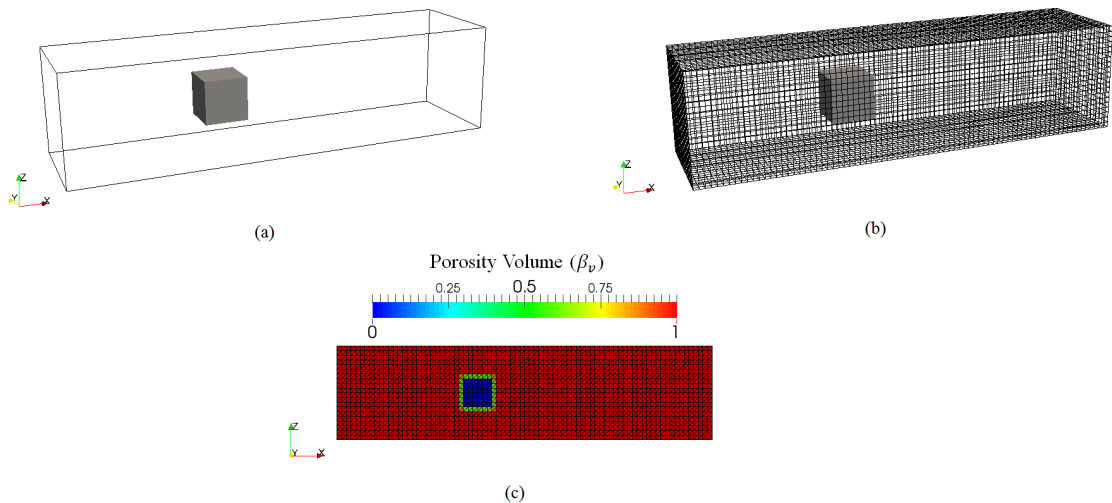


Figure 16 – Parametrisation of the cube .stl CAD file used as the bluff body in the fluid flow simulation using the 3D Euler solver. (a) Cube (CAD file) in the computational domain. (b) Uniform hexahedral cells of the computational domain. (c) Porosity field calculated by the GJK algorithm as shown in the central plane of the computational domain

In Figure 16 (c) the dark blue region represents the cube geometry. The computational cells in this region were assigned zero porosity. Hence, the flow is not allowed in this region of the computational domain. The red part represents the region avail-

able to the flow. The computational cells coloured between dark blue and red present intermediated porosity values ( $0.0 < \beta_v < 1.0$ ).

The air flow with inlet velocity of 160 m/s was evaluated. The inflow boundary conditions was set on the left boundary  $x$ . The outlet boundary condition, with pressure gradient equal to zero, was considered at the right boundary  $x$ . The other boundaries of the computational domain were set as wall, where zero normal fluxes of mass, momentum and energy are imposed. This strategy to set the boundary conditions was also applied to all simulations using the PFS code. Table 4 presents the values of solver parameters applied in this simulation and the results are shown in Figure 17.

Table 4 – Solver parameters used in the simulation of a flow over a parametrized squared geometry by using the 3D Euler solver.

Parameter	Value
Fluid	0 (air)
Inlet Velocity	160 m/s
CFL	0.01
Smooth Factor	0.05
Convergence Criterion	$10^{-4}$

Figure 17 shows that the velocity profile is similar to what is expected for a fluid flow over an object. The velocity increases where the area of flow decreases. By considering the PDR model, Figure 17 (a) shows that there is no flow in the computational domain region set with porosity values equal to zero (dark blue square in Figure 16 (c) ). It is also verified that the presence of the box blocks the flow and produces a re-circulation zone (17 (b)) . The stagnation zone is observed in front of the cube.

Once the code was able to solve the Euler equation in a three-dimensional porous mesh for flow around a squared box, a series of tests were performed to verify the influence of the simulation parameters in the results.

### 5.1.3 Simulation Parameters Analysis

In Chapter 4, the structure of PFS code was introduced. It was reported that the code runs under the unsteady condition and the time-step is obtained by the relation between mesh size and velocity, being adjustable by CFL number. It was also pointed out that an additional artificial viscosity must be included to reduce the numerical oscillations provided by the central difference scheme. Thus, a series of tests were performed to evaluate the influence of these parameters on the numerical solution.

The cubic parametrised geometry shown in Figure 16 was used here with the same fluid flow condition (air flow with 160 m/s). In all tests, the velocity profile was

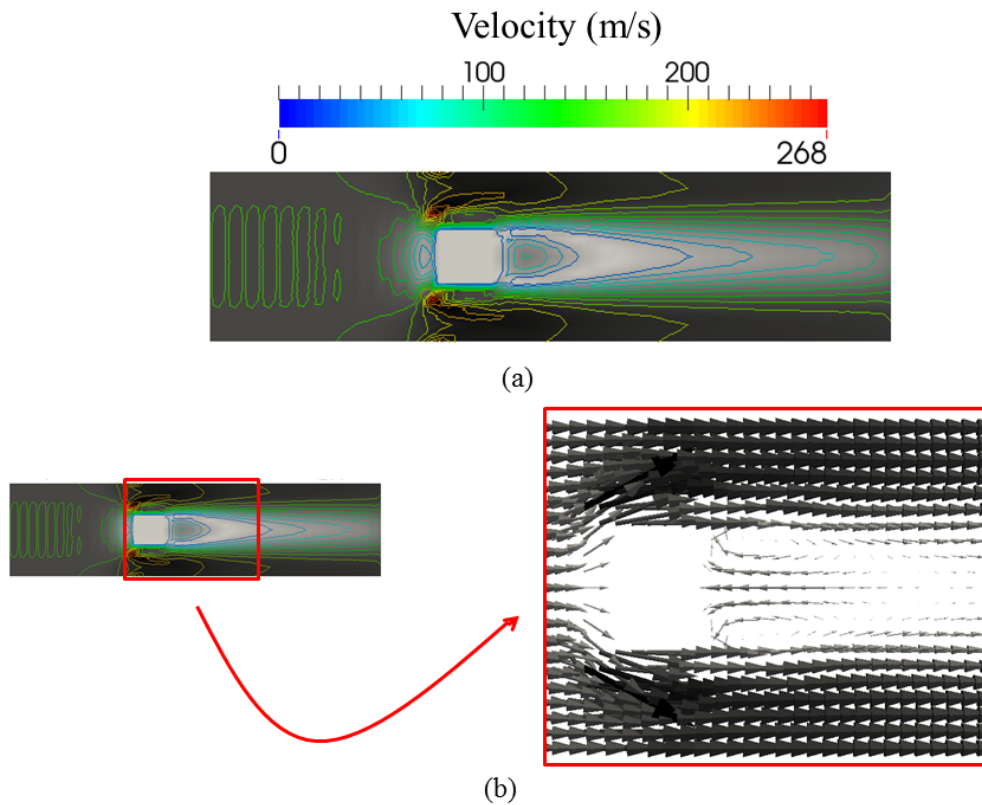


Figure 17 – Simulation of a flow over a parametrized squared geometry by using the 3D Euler solver: velocity field (a) and recirculation zone behind the geometry (b).

evaluated at a distance of 1.0 m downstream the squared box. The number of time-steps to achieve the steady condition and the computational time were also analysed.

### 5.1.3.1 Mesh Test

To evaluate the influence of the mesh on the numerical solution, 3 different sizes of computational cells were considered: 0.3m (Mesh 01), 0.15m (Mesh 02) and 0.075m (Mesh 03) of length. In Mesh 03, the squared geometry could be solved by 3.3 computational cells, 6.7 cells were applied for the intermediate mesh Mesh 02 and for the last mesh (Mesh 03) slightly more than 13 cells were used. The CFL number and the smooth factor were taken as 0.01 and 0.05 (Table 4). Figure 18 shows the result of the simulation.

It is possible to observe in Figure 18 (a) that the velocity profile is similar for all three considered meshes. The values obtained by using the intermediary and refined meshes (Mesh 02 and Mesh 03, respectively) are very close, while the coarse mesh (Mesh 01) under-predicts the flow velocity in the region behind the squared box. In Figure 18 (b) it is verified that reducing the computational cell size greatly increases the number of iterations to achieve the steady state flow (with a maximum residual equal to  $10^{-4}$ ).

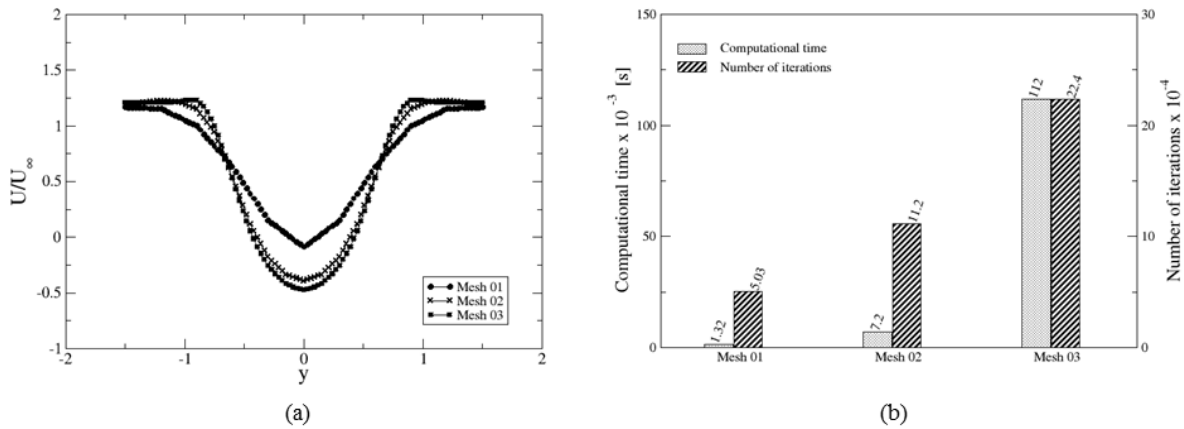


Figure 18 – Mesh test applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver: Mesh 01 represents the coarse mesh (computational cells with 0.3m of length), Mesh 02 is the intermediary mesh (computational cells with 0.15m of length) and Mesh 03 is the refined mesh (computational cells with 0.075m of length). (a) Velocity profile at the distance equal to 1 meter behind the squared box. (b) Number of iterations and computational time

When reducing the computational cell size, the time-step is reduced and the flow variables take a smaller increment in each iteration. Furthermore the number of grid points also increases and the computational time to complete an iteration is longer.

### 5.1.3.2 CFL Analysis

Different values of Courant number (CFL) were also considered to understand the influence of this parameter on the numerical solution of the developing code: 0.05, 0.01 and 0.005. In this case the mesh with computational cell size equal to 0.1 m and smooth factor 0.05 was adopted. The results of such analysis are presented in Figure 19.

Figure 19 shows (a) that, in this case, the CFL number has no influence on the final solution that represents the steady state condition of the flow. Velocity profile taken at a distance equal to 1m downstream the box is equal for the simulations with three different CFL numbers. However, CFL greatly influences the number of iterations and computational time, as observed in Figure 19 (b). Similar to reduce the computational cell size, when reducing CFL, the time-step also reduces, therefore in each iteration the flow variables take a smaller increment and more iterations are necessary to achieve the steady state.

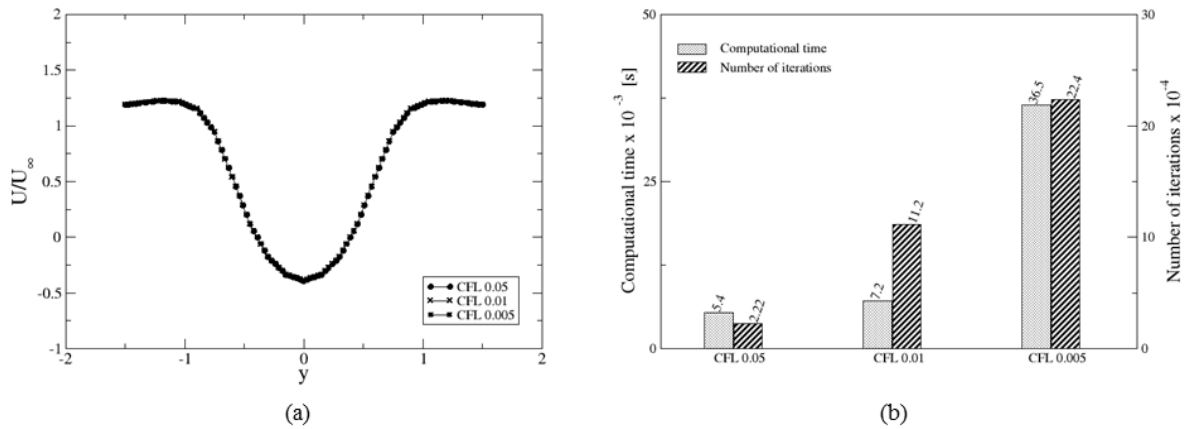


Figure 19 – CFL test applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver.

### 5.1.3.3 Smooth Factor Analysis

Considering mesh with computational cells size equal to 0.1 m and CFL 0.05, different values of smooth factor were also analysed ranging from 0.025 - 0.1. Figures 20 and 21 present the results of this analysis.

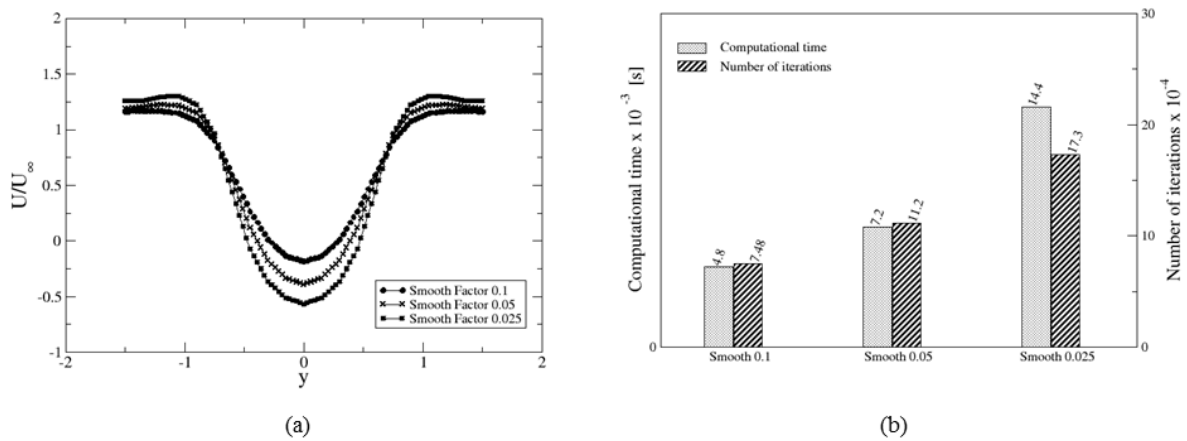


Figure 20 – Numerical test with different values of smooth factor applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver.

Figure 20 shows that when using larger values of smooth factor, the number of iterations and computational time to achieve the steady flow is lower than when smaller values of smooth factor are considered.

Analysis of Figure 21 shows that when small values of smooth factor are applied oscillations on velocity field are noticed, particularly in front of the box (Figure 21 (a))

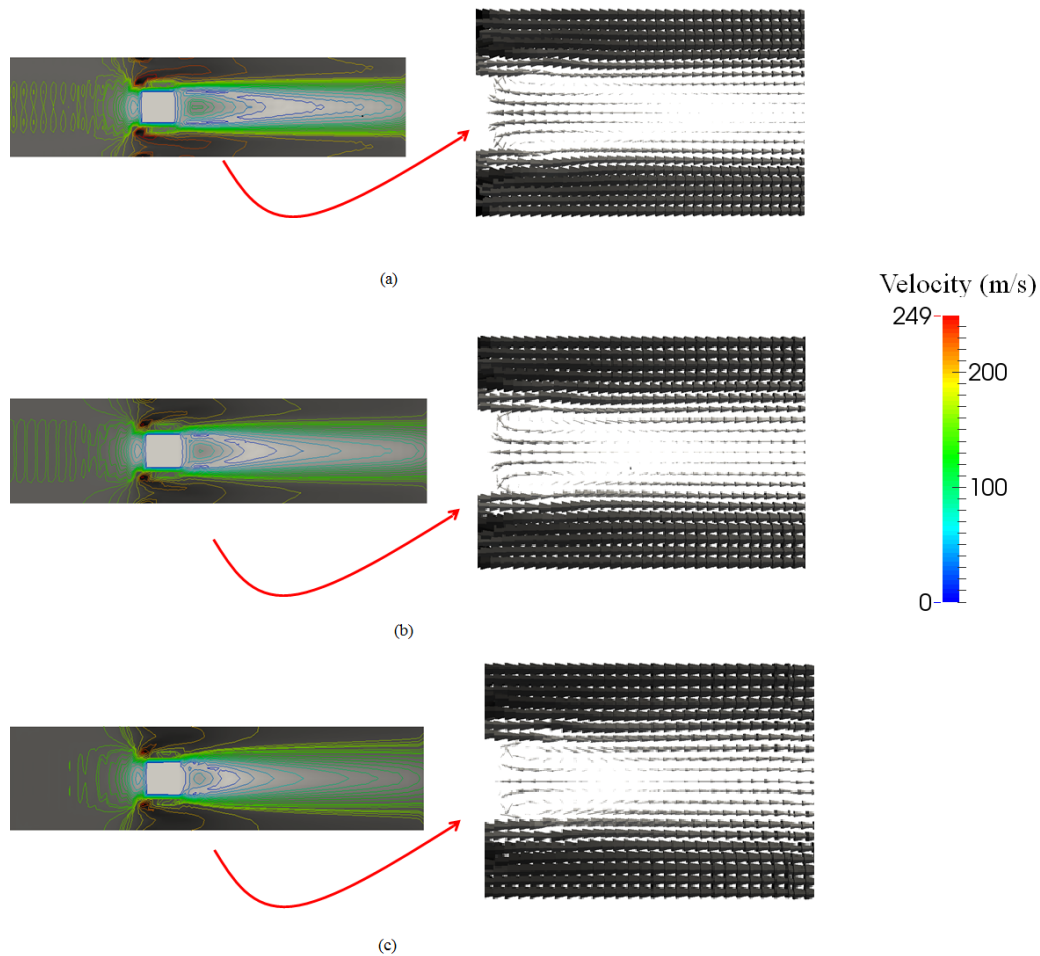


Figure 21 – Qualitative analysis of different values of smooth factor applied to a simulation of a flow over a parametrised squared geometry by using the 3D Euler solver. Smooth values: 0.025 (a), 0.05(b) and 0.1 (c)

). Such oscillations are reduced when higher values of smooth factor are applied. As previously discussed in Section (4.2.3), the artificial viscosity is applied to reduce the oscillations commonly found on numerical solution when the central difference scheme is used. When higher values of smooth factor are used, the effect of artificial viscosity is more effective, and the numerical oscillations present on the velocity field in Figure 21 (a) are reduced, as observed in Figure 21 (c). On the other hand, when introducing artificial viscosity, the physical behaviour of the phenomenon is also affected. As observed in Figure Figure 21 (c), the recirculation zone behind the squared box is smaller than in Figure 21 (a), where artificial viscosity is less pronounced.

The utilisation of artificial viscosity is always dodgy and it is a compromise between accuracy and stability.



### 5.1.4 Verification of the 3D Navier-Stokes code

The complete set of the Navier-Stokes equations was implemented and coupled with the  $k - \varepsilon$  turbulence model. To evaluate the code implementation at this step, the flow over a squared box, described in section 5.1.2, was simulated again considering the air flow with 100m/s at the entrance of the computational domain. Similar geometry and simulation conditions were reproduced in the commercial CFD code *ANSYS CFX 15.0*. Figure 22 and 23 present the comparison between these two numerical results.

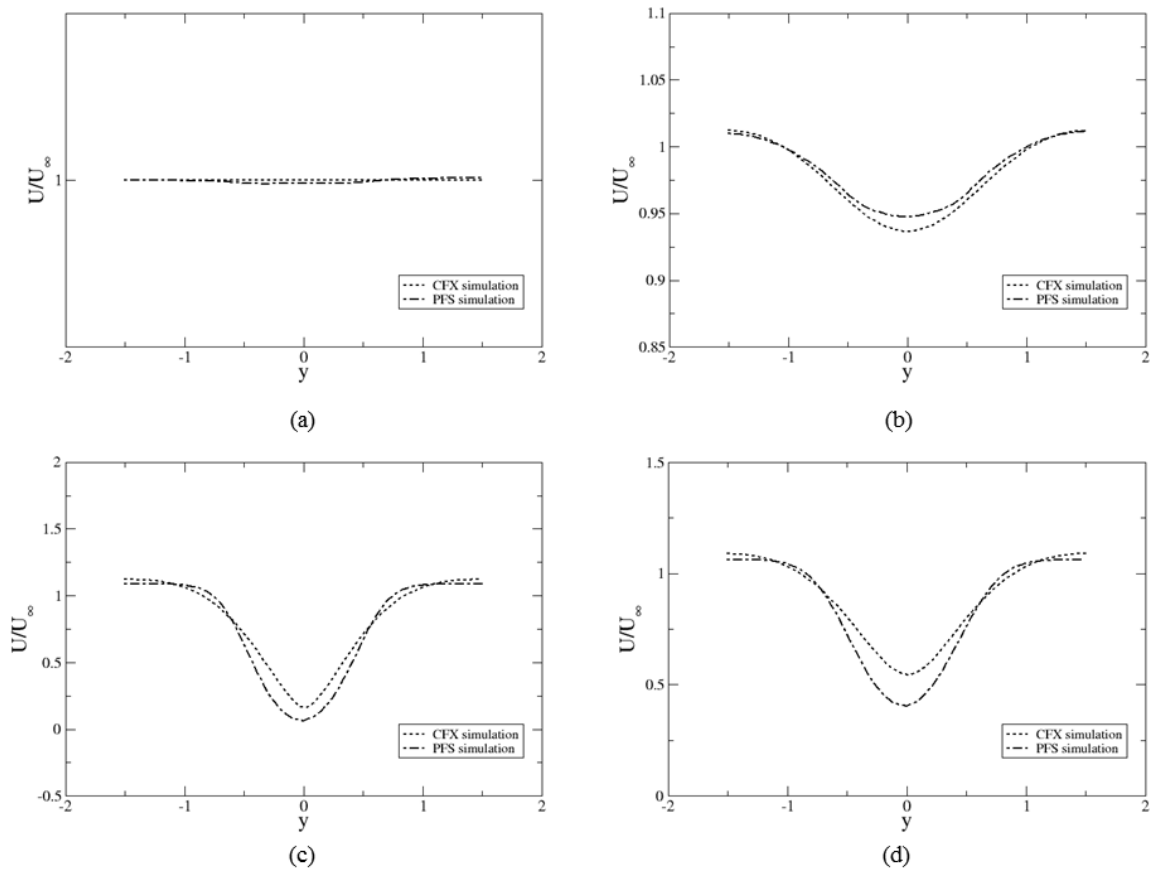


Figure 22 – Time mean velocity profile at different distances upstream and downstream the rectangular box. (a)  $x = 2\text{m}$  before the rectangular box; (b)  $x = 1\text{m}$  before the rectangular box; (c)  $x = 1\text{m}$  downstream the rectangular box; (d)  $x = 2\text{m}$  downstream the rectangular box.

In Figure 22 the velocity profile is analysed at two distances upstream ( (a) and (b)) and two distances downstream ( (c) and (d)) the rectangular box. Additionally, in Figure 22 the velocity profile is taken at the centreline downstream of the squared box. In both cases, the numerical results provided by the developed code describes velocity profiles similar to the ones obtained when using the commercial code.

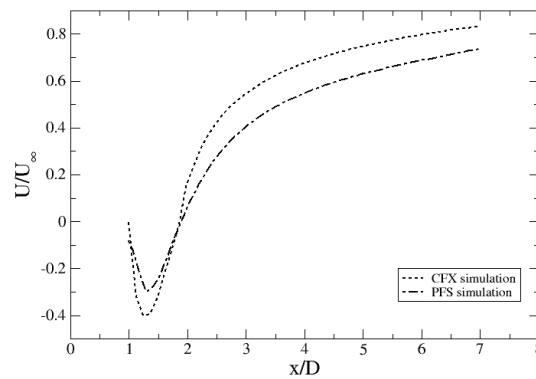


Figure 23 – Time mean velocity profile at the centreline downstream of the rectangular box.

## 5.2 Analysis of turbulent flows in simplified geometries

Once the code was verified and able to solve the set of Navier-Stokes equations coupled with the  $k-\varepsilon$  turbulence model, traditional cases of turbulent flows were simulated based on literature review.

### 5.2.1 Flow over a backward facing step

In many engineering applications it is necessary to understand the physics of the phenomenon of separation and reattachment of turbulent flow. Such a phenomenon can be analysed in a simple way through the study of turbulent flow over a backward-facing. To evaluate how the PFS code reproduces the separation and reattachment of a turbulent flow, a turbulent flow over a step was considered.

Based on an experimental study (KIM *et al.*, 1980) and a numerical simulation (YOO *et al.*, 1989), a fluid flow at Reynolds number equal to 44,000 was simulated over a step with 10 mm of height ( $H$ ) and 40 mm of length. The computational domain used in this case comprises a rectangular box with 340 mm in  $x$  direction, 3 mm in  $z$  direction and 10 mm in  $y$  direction. Air flow with velocity of 70m/s ( $U_\infty$ ) was imposed at the entrance of the computational domain. Figure 24 shows the schematic representation of such study.

The solver parameters can be found in Table 5.

Initially the simulation was conducted considering three different meshes: the first one with hexahedral cells measuring 1.5 mm of length, the second mesh with cells length of 1.0mm and the third one with cells of 0.5 mm of length. Figure 25 shows the mean velocity profile in the region downstream of the cylinder at a distance  $x/H$  equal to 7. The values were taken along a vertical line. Mesh 01 represents the mesh with computational cells with 1.5 mm of length (the coarse mesh), while Mesh 02 represents

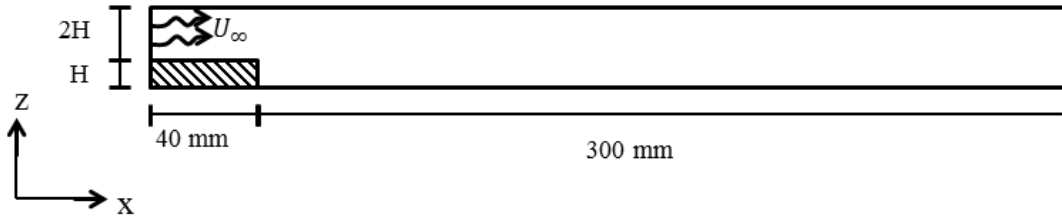


Figure 24 – Schematic representation of the simulation of a turbulent flow over a backward-facing step.

Table 5 – Solver parameters used in the simulation of a flow over a backward facing step by solving the set of Navier-Stokes equations.

Parameter	Value
Fluid	0 (air)
Inlet Velocity	70 m/s
CFL	0.5
Smooth Factor	0.05
Convergence Criterion	$10^{-4}$

the mesh with computational cells with 1.0 mm of length (intermediary mesh) and Mesh 03 represents the mesh with computational cells with 0.5 mm of length (refined mesh).

Figure 25 shows that the results provided by Mesh 01 are very different from those obtained using Mesh 02 and Mesh 03. The profiles yielded by Mesh 02 and 03 are slightly different for the region just after the step (that represents the re-circulation zone), but they become similar for the region of the developed flow. Because of the complexity of the flow at the re-circulation zone, the results provided in such a region are very sensitive to the mesh parameters, so the refined mesh provided better results than the coarse meshes at the re-circulation zone.

Based on the analysis of Figure 25, a mesh with computational cell size equal to 1.0 mm was chosen for the study of turbulent flow over a backward facing a step. The results of the simulation are presented in Figures 26 to 28. It is possible to observe the presence of a re-circulation zone downstream the step which characterises flow separation in turbulent flows over a step . An analysis here is to compute the reattachment length. Experimental study conducted by Kim *et al.* (1980) presents a mean reattachment distance of  $x/H = 7 \pm 1$  for turbulent flow over backward-facing step. The PFS simulation provides a reattachment length of  $x/H = 8.5$  which is close to the value observed experimentally.

Figures 27 and 28 show the local mean velocity profiles and turbulence intensity

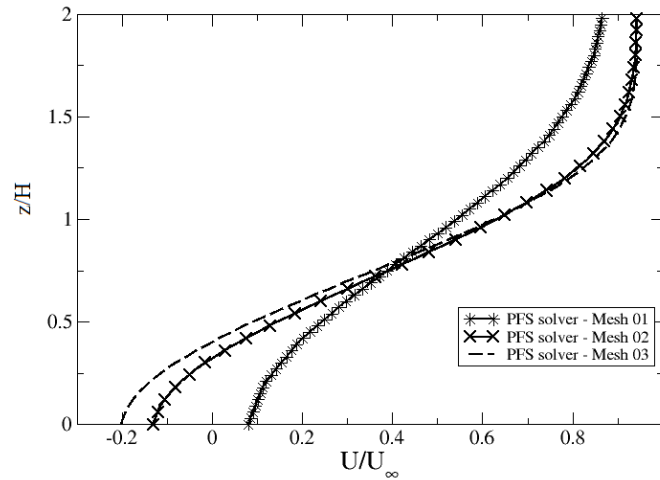


Figure 25 – Time mean velocity profile at the distance  $x/H = 7$  downstream the step for three different sizes of mesh. Mesh 01 is the coarse mesh (computational cells size = 1.5mm); Mesh 02 is the intermediary mesh (computational cells size = 1.0mm) and Mesh 03 is the refined mesh (computational cells size = 0.5 mm).

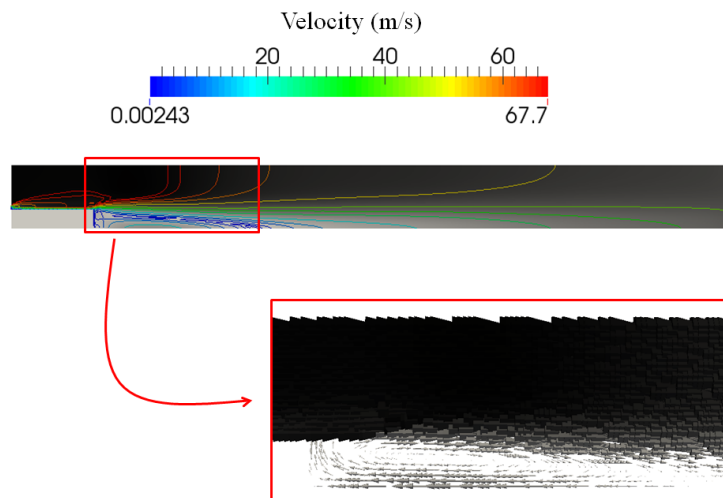


Figure 26 – Simulation of a turbulent flow over a backward-facing step. Velocity field (a) and velocity vectors at the recirculation zone (b)

at four different locations from the step.

Figure 27 shows that the velocity profiles provided by PFS code are similar to the experimental data at the re-circulation region (Figure 27 (a) and (b) ), which indicates that PFS code is able to represent the velocity behaviour at the separation zone of a turbulent flow over a step. However, such an agreement is not observed for the region

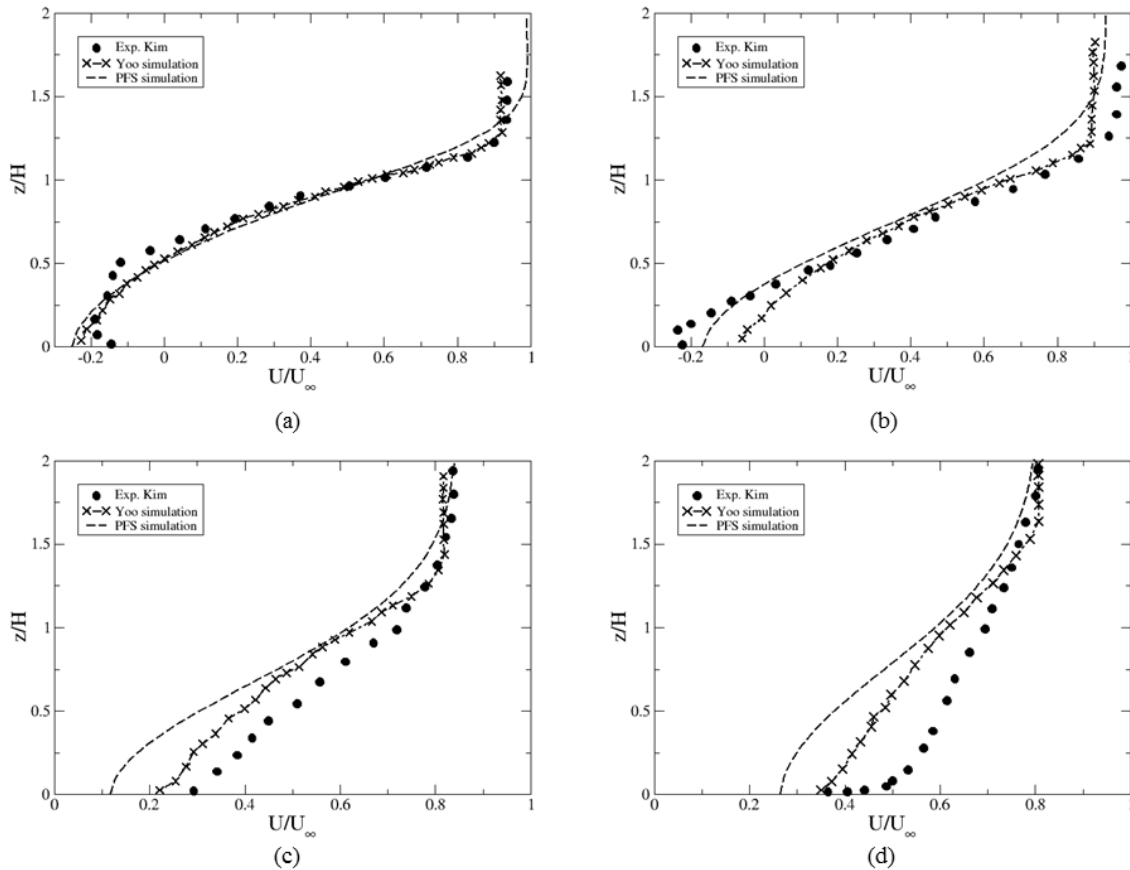


Figure 27 – Non-dimensional velocity profile of a turbulent flow over a backward-facing step at  $x/H = 2.67$  (a);  $5.33$  (b);  $9.78$  (c) and  $16$  (d).

far from the step (Figure 27 (c) and (d) ). For distances where  $x/H = 9.78$  and  $16$  the PFS simulation results differs from experimental data especially in the region near the floor. Similar profiles are observed in other numerical simulation using  $k - \varepsilon$  turbulence model (YOO *et al.*, 1989). The lack of agreement in this region suggests that in PFS code the boundary layer modelling may be enhanced.

Figure 28 shows that the PFS code over-predicts the turbulence intensity for flow over a backward facing. The same behaviour is observed in the numerical study of Yoo *et al.* (1989) that also uses the  $k - \varepsilon$  equations to model turbulence. At the region closer to step ( $x/H = 2.33$ ) the PFS results over predicts almost 2 times the turbulence intensity provided by the other  $k - \varepsilon$  simulation but such difference tends to reduce as the flow moves away from step ( $x/H = 5.88$  and  $8.53$ ). A different behaviour is observed for the region far from the step ( $x/H = 15.67$ ) where the PFS simulation under-predicts the turbulence intensity.

An important consideration here is that because PFS code solves all geometry as porous media, the viscous boundary layer is not adequately modelled. There is no

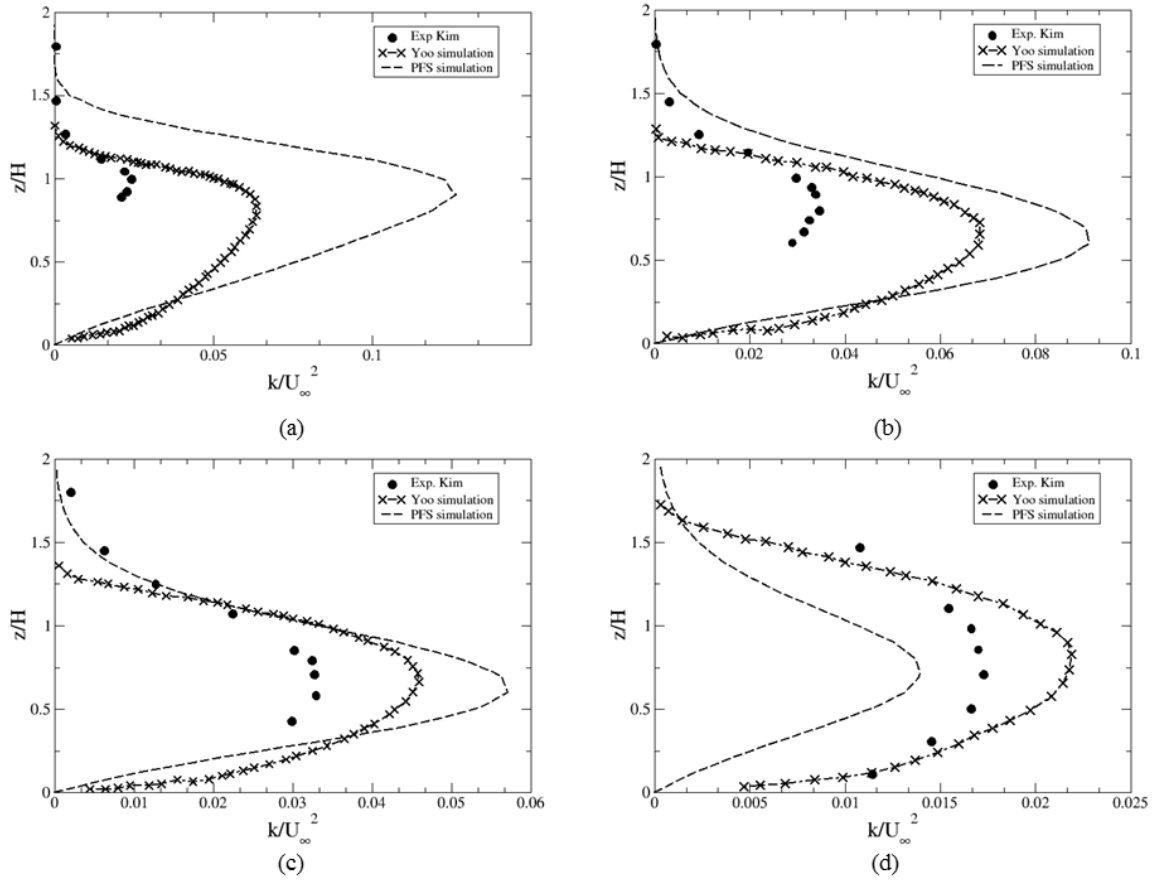


Figure 28 – Turbulence intensity of a turbulent flow over a backward-facing step at  $x/H = 2.33$  (a); 5.88 (b); 8.53 (c) and 15.67 (d).

implementation of wall boundary conditions for velocity profile and therefore the viscous stress at the region near wall can take greater values. This behaviour affects both diffusive terms (for all quantities) and the production term of turbulent kinetic energy. The influence on diffusive terms can be lower in problems where the convection is predominant (as is this study of turbulent flow over a step).

### 5.2.2 Flow past a rectangular cylinder

To evaluate how the PFS code describes the fluid flow around bluff bodies, a squared cylinder with length of 10 mm in each direction ( $z$ ,  $y$  and  $z$ ) was used as geometry under study and air flow was considered. Therefore, the flow at Reynolds number equal to 22,000 was simulated. This value of Reynolds number is similar to that considered in the experimental work conducted by (LYN *et al.*, 1995) and (DURAO *et al.*, 1988) and in the numerical simulation (RODI, 1997) of flow around a rectangular cylinder.

The computational domain used in this study comprised a rectangular box with 200 mm in  $x$  direction and 80 mm in  $y$  and  $z$  directions. The rectangular cylinder

was placed into the computational domain by considering the distance of 40 mm from y and z direction and 30 mm from x direction. The schematic representation of this analysis can be visualized in Figure 29.

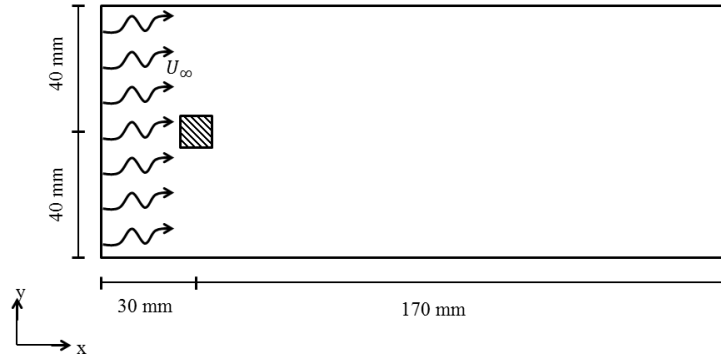


Figure 29 – Schematic representation of the geometry used in the study of a flow around a rectangular cylinder

As in the previous analysis for simulation of turbulent flow over a step, the simulation was conducted considering three different meshes: a coarse mesh (Mesh 01) with hexahedral cells with 1.5 mm of length, an intermediary mesh (Mesh 02) with cells length of 1.0mm and a refined mesh (Mesh 03) with computational cells size of 0.5 mm. Figure 30 shows the time mean velocity profile along the centreline of the square cylinder for three different mesh sizes. The solver parameters used in these simulation were the same applied to the previous study of a flow over a step (CFL number equal to 0.5 and smooth factor of 0.05).

Figure 30 shows that the results provided by the three meshes are slightly different for the region right behind the cylinder (that represents the re-circulation zone), but they become very similar for the region far from the square cylinder (when the distance from the cylinder is greater than 4 times the square length). It is also verified that the profile provided by the coarse mesh (Mesh 01) presents some oscillations. Such behaviour can be explained by the presence of insufficient artificial viscosity to minimize the numerical oscillations inherent to the central difference discretisation method used in PFS solver. The amount of artificial viscosity depends on many characteristics of the simulation such as the velocity gradient and the property increment in each iteration. In all three simulated meshes the velocity gradient were the same (Reynolds number equal to 22,000). However, the property increment increases as the mesh increases, so that the amount of artificial viscosity used in the simulation of the refined mesh could be not enough to minimize the oscillations when a coarse mesh was used.

Based on the analysis of Figure 30, a mesh with computational cell size equal to 1.0 mm was chosen for the study of the flow around a squared cylinder. Figure 31

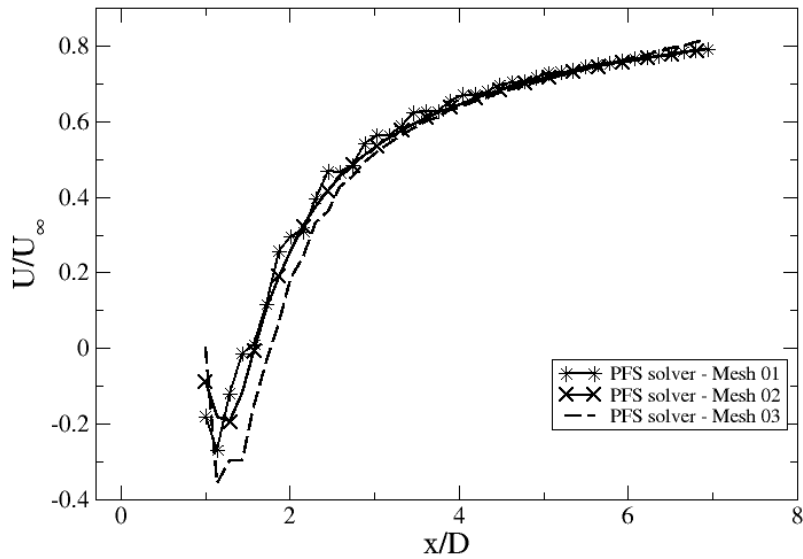


Figure 30 – Time mean velocity profile along the centreline of a square cylinder for three different sizes of mesh. Mesh 01 is the coarse mesh (computational cells size equal to 1.5mm); Mesh 02 is the intermediary mesh (computational cells size equal to 1.0mm) and Mesh 03 is the refined mesh (computational cells size equal to 0.5 mm).

shows the result of the simulation. The velocity profile is similar to what is expected for a turbulent flow over a rectangular cylinder. The velocity increases where the area of the fluid flow decreases. By considering the PDR model, there is no flow in the computational domain region set with porosity values equal to zero. The presence of the squared cylinder blocks the fluid flow and produces a re-circulation zone. A resistance region is observed in front of the rectangular cylinder.

Figure 32 shows the comparison between the PFS simulation results by using the intermediary mesh (MESH 02) and experimental data provided by two different authors: Lyn *et al.* (1995) and Durao *et al.* (1988). Both works have investigated turbulent flows over rectangular cylinder, but Lyn *et al.* (1995) used a Reynolds number equal to 22,000 in the experiments, while for Durao *et al.* (1988) experiment the Reynolds number was equal to 14,000. The non-dimensional velocity and turbulence intensity profiles provided by PFS simulation are similar to the experimental data.

Concerning the non-dimensional velocity present in Figure 32 (a), the PFS results are close to both experimental data in the region near the cylinder, but is slightly different from Lyn *et al.* (1995) experimental data as the flow moves away from the cylinder. The turbulence intensity is visualized in Figure 32 (b). Here, the turbulence intensity provided by PFS is a little different from the experimental data especially in the



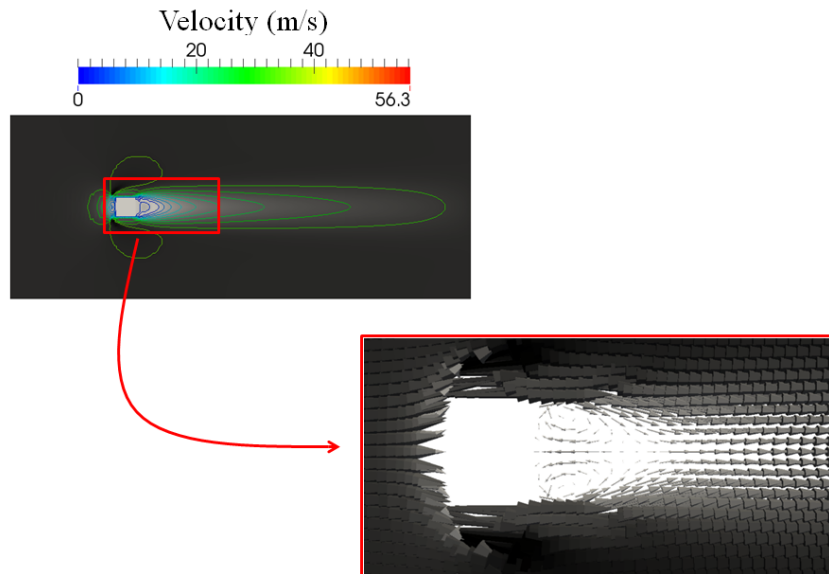


Figure 31 – Velocity field of a turbulent flow by considering the geometry of a rectangular cylinder as a porous media.

region near the cylinder (that represents the recirculation zone). Such a difference can be explained by the drawbacks in  $k - \varepsilon$  turbulence model in dealing with recirculation zones.

### 5.2.3 Flow over a mounted rectangular cylinder

The analysis of turbulent flow over a mounted cylinder was also considered. Following Rodi (1997) study, the same squared cylinder analysed in Section 5.2.2 was used here and placed on the lower wall (floor) of a wind channel (computational domain).

The channel height was set as the double of cube size ( $H$ ) and the calculation domain was built considering the distances of  $3 \times H$  upstream,  $10 \times H$  downstream and  $7 \times H$  laterally of the rectangular cylinder. Following the mesh test analysis provided in section 5.2.2, the size of the computational cells was chosen to be equal to 1.0 mm, so the cube geometry could be represented by at least 10 cells (similar strategy was considered in the other PFS simulations). A Reynolds number of 40,000 was considered. The inlet velocity was set to 53 m/s at the entrance of the computational domain. Figures 33 presents the schematic representation of this study.

Simulation results are observed considering streamlines and vector fields in a symmetry plane (Figure 34) and near the channel floor (Figure 35). Both views shows the separation zone at the region behind the cylinder. Considering the symmetry plane, the extension of the recirculation zone obtained by numerical simulation was close to  $x/H = 2.0$  while experimental study (RODI, 1997) provides a recirculation zone extension

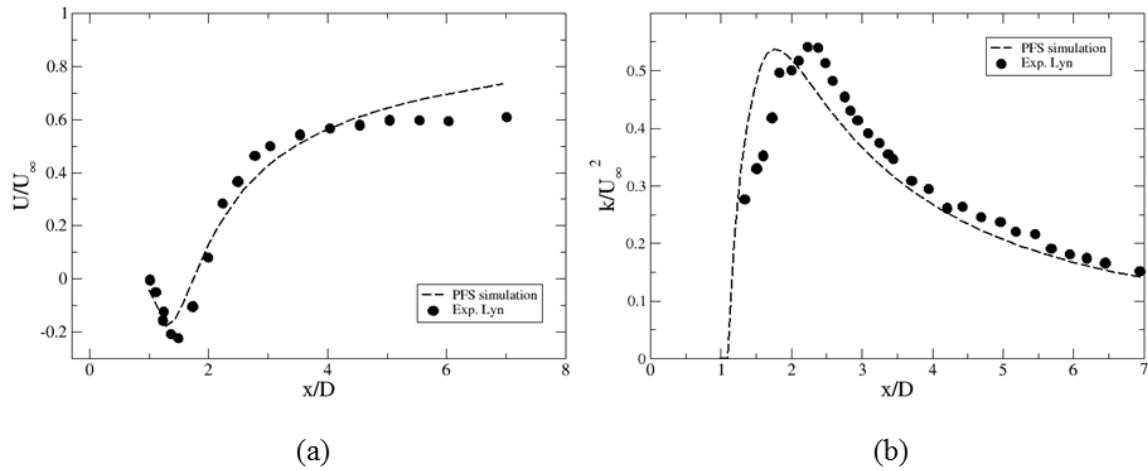


Figure 32 – Comparison between PFS simulation results and experimental data along the centreline of a square cylinder. Axis "x" presents the non-dimensional distance from the cylinder, while axis "y" presents the the non-dimensional velocity (a) and turbulence intensity (b)

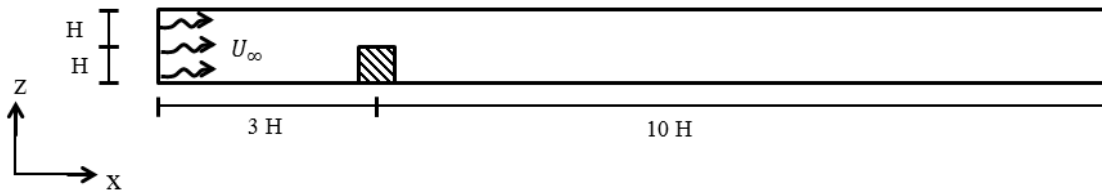


Figure 33 – Schematic representation of the simulation of a turbulent flow over a mounted squared cylinder.

equal to  $x/H = 1.6$ . Thus, as observed for previous studies of flow with separation zone (section 5.2.1), the results obtained by the developed solver over-predicts the extent of the separation region behind an obstacle (20%). Such a lack of agreement is also observed in other RANS simulations, where the standard  $k - \varepsilon$  turbulence model can predicts the separation region (35%) (RODI, 1997).

Figure 36 compares profiles of streamwise velocity  $U$  and turbulent kinetic energy  $k$  at different downstream locations in the symmetry plane. Here, the dimensionless distances  $x/H$  were obtained by considering the point  $x = 0.0$  at the beginning of geometry.

By analysing the velocity profiles (Figure 36 (a) ), it is observed that the present numerical simulation is capable to describe the overall flow behaviour provided by experimental data (RODI, 1997) at the three considered distances, although the lack of agreement is verified at the regions close to the channel walls. As previously discussed in

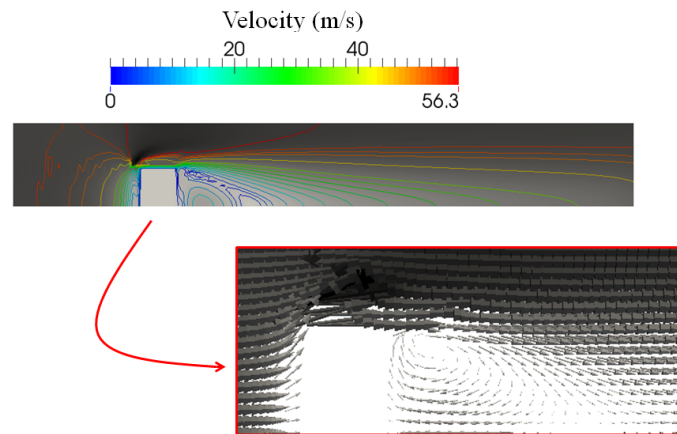


Figure 34 – Velocity field of a turbulent flow over a mounted cylinder in a longitudinal symmetry plane.

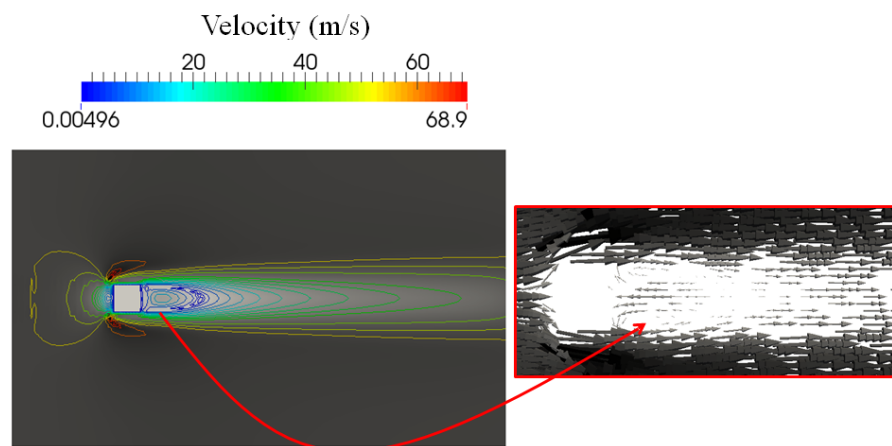


Figure 35 – Velocity field of a turbulent flow over a mounted cylinder near the channel floor.

the turbulent flow over a step, the developed solver does not account for the viscous effect near the wall. It is clear that an enhanced boundary layer model for porosity parametrisation is required.

Concerning the turbulent kinetic energy profiles given in Figure 36 (b), it is also observed a good agreement between simulation results and experimental data (RODI, 1997), specially for the two initial distances ( $x/H = 0.5$  and  $x/H = 1.0$ ). The third turbulence profile ( $x/H = 2.0$ ) deals with the reattachment region and numerical findings present a slight different behaviour compared to experimental data.

## 5.2.4 Flow around cylinders

Geometries with circular shape were also analysed when evaluating PFS code to describe turbulent flows behaviour. In this case, a set of experiments conducted by

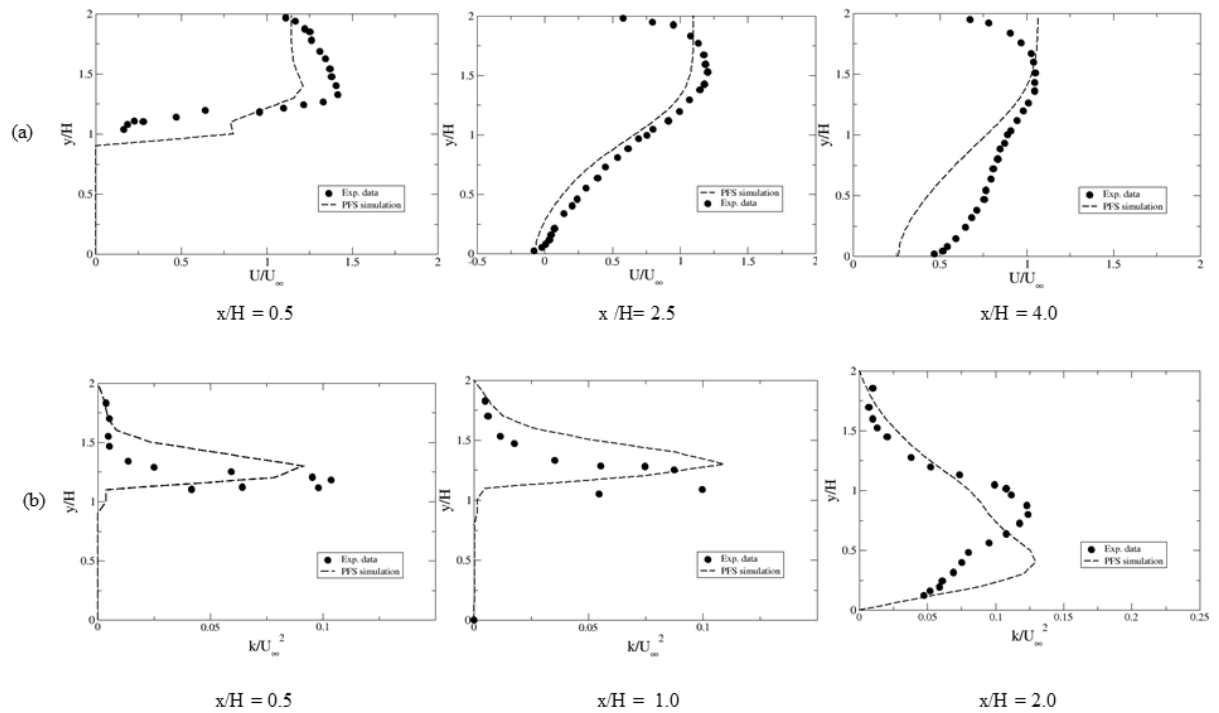


Figure 36 – Velocity and a turbulence intensity profiles in a symmetry plane at different distances from the beginning of the mounted squared cylinder.

British Gas and CMR (Cristian Michelsen Research) and reported by Arntzen (1998) was considered.

The geometries described in this study have been used to validate different CFD codes ((ARNTZEN, 1998), (VIANNA; CANT, 2010)) and consist of one single pipe (with 0.5m of diameter) and a array of five pipes (each pipe with 0.17m of diameter). For both cases, the computational domain was built measuring  $12.0\text{m} \times 3.0\text{m} \times 1.0\text{m}$  (in x, y and z directions, respectively). The computational mesh was built for each case considering that at least 10 computational volumes are used to describe the geometry. Figure 37 shows the schematic representation of these studies.

The simulation of a turbulent flow was conducted considering air flowing at of 25 m/s at the entrance of computational domain (x left hand side) for both cases and the results are presented in Figure 38. The main features of a turbulent flow over an obstacle are also verified here. The stagnation region is observed in the front of the single cylinder and the bank of cylinders. Recirculation zones are also presented downstream the obstacles in both cases.

The velocity profile and the turbulent velocity fluctuation were analysed at the region downstream the cylinders in both cases. The velocity profile was taken as the mean velocity in flow direction while the turbulent velocity fluctuation was obtained from the

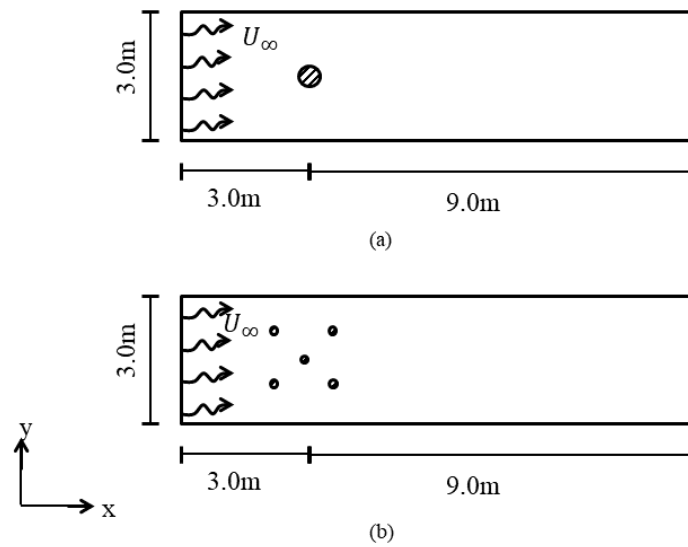


Figure 37 – Schematic representation of the simulation of flow around cylinders - CMR tests. (a) one single pipe with 0.5m of diameter; (b) bank of five pipes, each one with 0.17m of diameter.

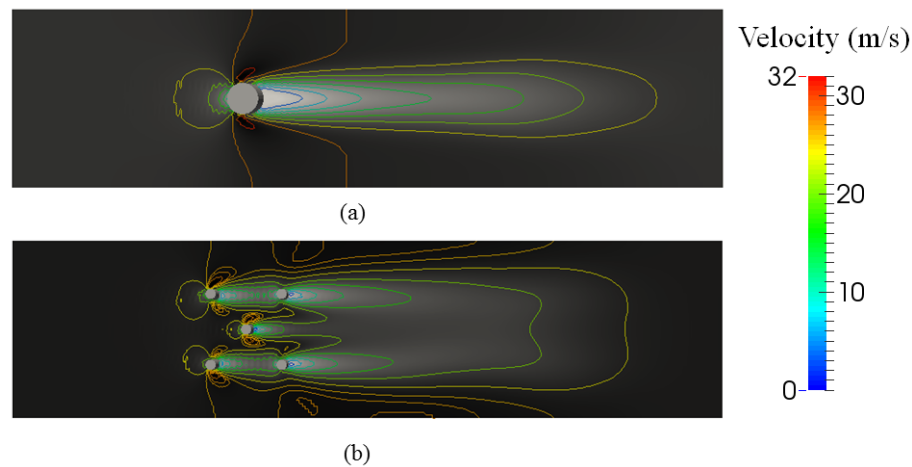


Figure 38 – Velocity flow field for CMR experiments. (a) one single pipe with 0.5m of diameter; (b) bank of five pipes, each one with 0.17m of diameter.

turbulent kinetic energy ( $k$ ). For one single pipe (Figure 37 (a)), the results were taken 1.0 m downstream the pipe, while for the bank of five pipes, the results were obtained 0.66 m downstream the last row of pipes. The comparison between experimental data and numerical results provided by PFS are presented in Figure 39.

The PFS simulation is capable of reproducing the overall trend of mean velocity and turbulence intensity at the considered flow region. Concerning the velocity profile, the numerical results slightly differ from the experimental data at the wake region which indicates that, in these cases, the porosity model offers a higher resistance to the flow. When analysing the turbulent velocity fluctuation, it is verified that numerical results

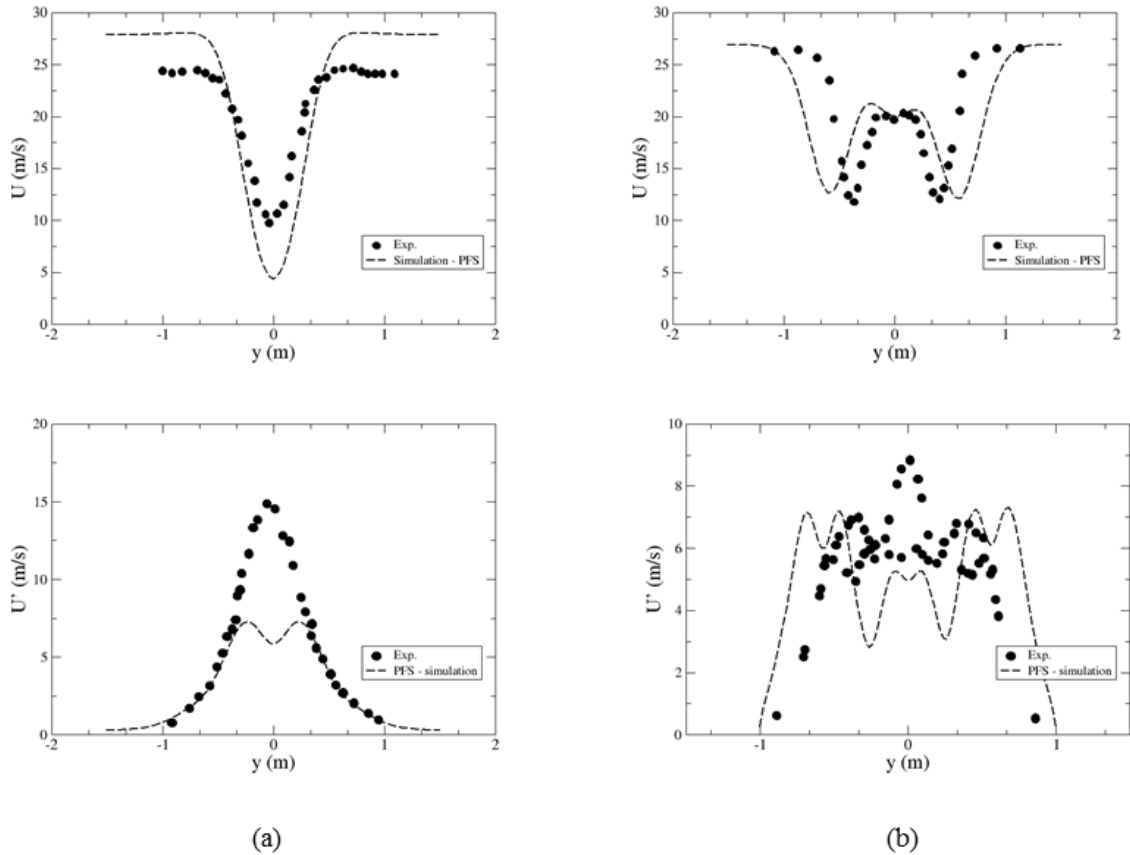


Figure 39 – Comparison between PFS simulation and experimental data (ARNTZEN, 1998) considering the mean velocity profile and the turbulent velocity fluctuation at the region downstream the considered geometries: (a) one single pipe with 0.5m of diameter; (b) bank of five pipes, each one with 0.17m of diameter.

under-predict the turbulence intensity, which is more evident in the simulation with one single cylinder.

The lack of agreement between the numerical results and experimental data can be explained by the fact that in these cases, the porous region representing each cylinder is not circular as the real geometry. The porosity model is capable to filter and represent the overall shape of geometry but can not reproduce it exactly. Furthermore the turbulence model used at the developed code to account the effects of non-resolved obstacles (described in Section 3.6.2.1) is very simple and accounts mainly the turbulence production due to shear.

Although the differences between numerical and experimental data, the PFS results seems to be in accordance with others RANS simulations of turbulent flows using PDR method coupled with  $k - \varepsilon$  equations (ARNTZEN, 1998) (VIANNA; CANT, 2010)

(SAVILL; SOLBERG, 1994).

### 5.3 Analysis of turbulent flows in complex geometries

Once the developed model was evaluated for flows around simplified geometries, complex geometries have been considered. Two different scenarios of complex geometry were built in the FreeCad tool and simulations were performed using the ANSYS-CFX CFD tool and the PFS developed tool.

Some illustrative examples of the technique developed in the current research, to solve flow in complex geometries in a real scale, can be found in Appendix B.

#### 5.3.1 Turbulent cross-flow in a staggered tube bundle

The first case study of PFS simulation in complex geometries comprises the investigation of turbulent cross-flow in a tube bundle. In this case, a set of tubes with 2.54 mm of diameter ( $D$ ) was staggered positioned in a wind channel with dimensions of  $2500 \times 191 \times 191$  mm, as shown in Figure 40.

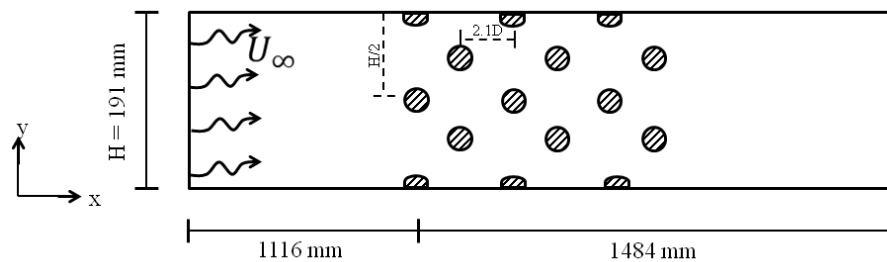


Figure 40 – Schematic representation of the simulation of turbulent cross-flow in a staggered tube bundle.

By applying the GJK algorithm, the geometric stl model was converted in a porous hexahedral mesh considering cells with size of 0.254 mm. Figure 41 shows the volume porosity obtained for each computational cell in a slice plane of the computational domain. A three-dimensional view of the tube bank represented by their porosity values is presented in Figure 41 (b). In this case, intermediate porosity values (between 0.0 and 1.0) were obtained for cells at the tubes contour.

An inlet air flow of 50 m/s was set at the entrance of the computational domain (x left hand side, as visualized in Figure 40). The simulation was performed considering the PFS parameters: Courant number (CFL) = 0.5; smooth factor = 0.1, and convergence criterion =  $10^{-4}$ . The scalar velocity field obtained as result of this simulation is presented in Figure 42.

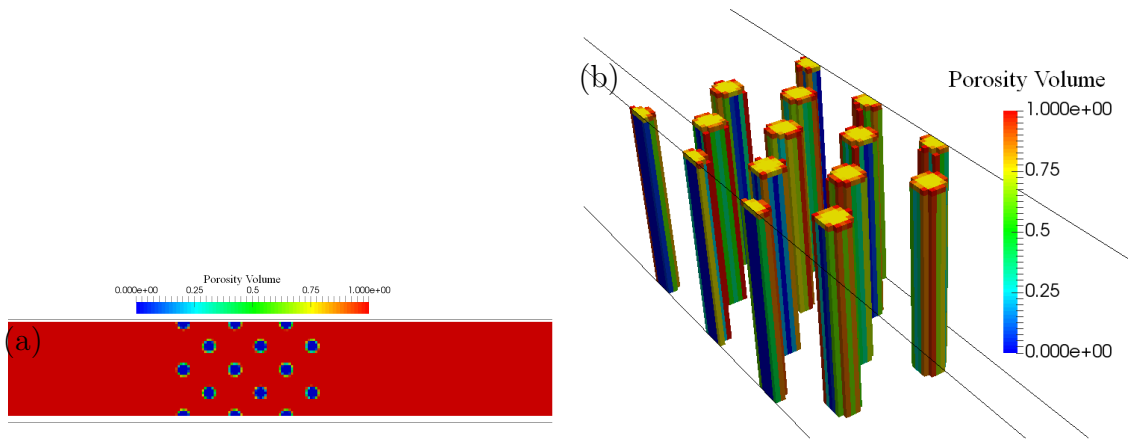


Figure 41 – Porous mesh obtained for a staggered tube bundle: slice plane view (a) and three-dimensional view (b).

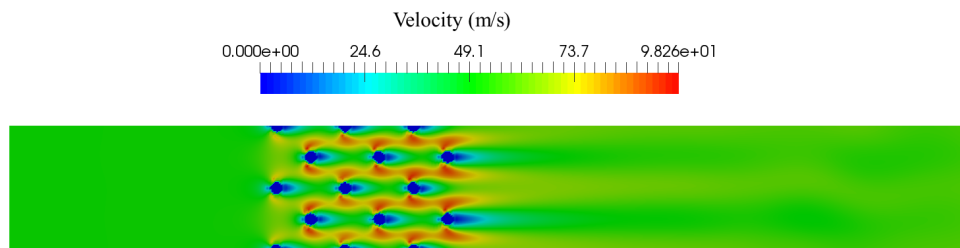


Figure 42 – Scalar velocity field obtained as result of the PFS simulation of turbulent cross-flow in a staggered tube bundle.

A simulation was also performed using the ANSYS-CFX commercial CFD tool. Similar flow and boundary conditions were considered, however, the CFX simulation uses the upwind advection scheme to solve the discretized equations.

The PFS results were compared with those ones provided by CFX simulation. Considering the origin of the coordinate system at the centre of the first cylinder, monitor lines were positioned at four different zones of the tube bundle:  $x/D = 1.25$ ;  $x/D = 3.35$ ;  $x/D = 5.45$ ; and  $x/D = 7.55$ . The velocity profile was taken at these lines for both PFS and CFX simulation and are presented in Figure 43. The flow behaviour provided by PFS are very similar to the one yields by CFX simulation.



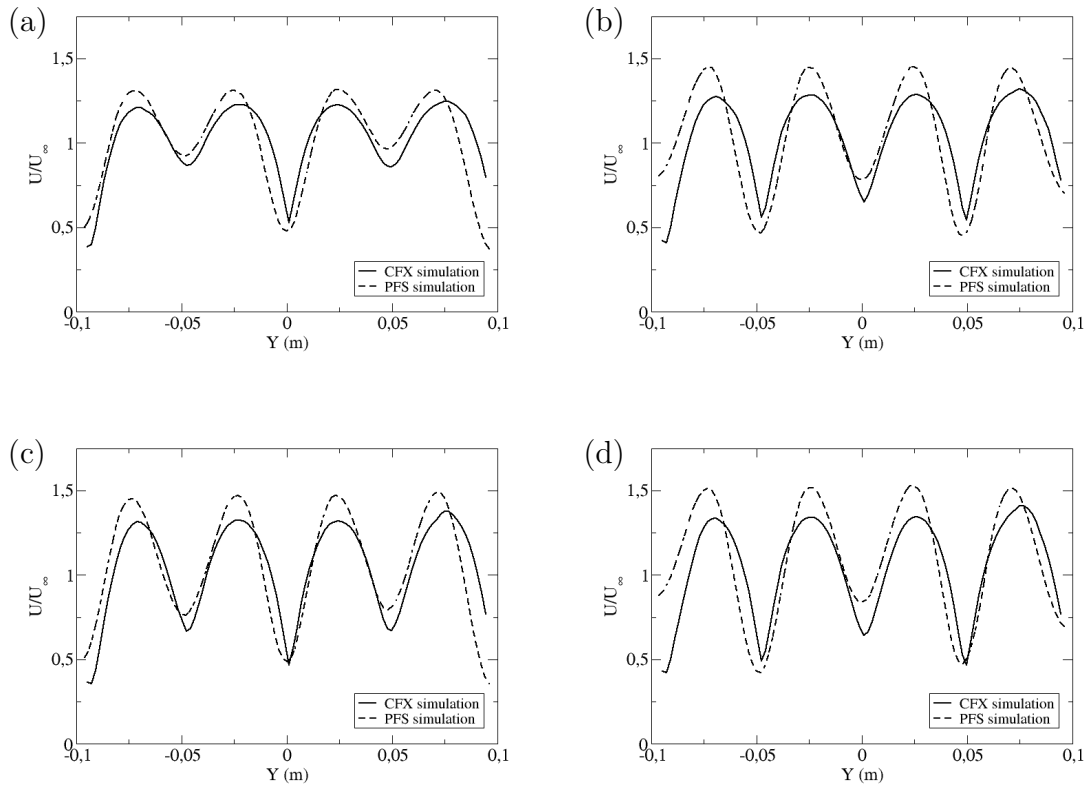


Figure 43 – Comparison of the simulation of a turbulent cross-flow in a staggered tube bundle by the developed PFS tool and ANSYS-CFX. Velocity profiles were taken at four different regions of the domain:  $x/D = 1.25$  (a);  $x/D = 3.35$  (b);  $x/D = 5.45$  (c); and  $x/D = 7.55$  (d).

### 5.3.2 Flow in an offshore module

The second investigation of flow in complex geometry comprises the ventilation analysis in an idealized offshore module. The geometric model used here can be visualized in Figure 44 (a) and comprises an oil platform with size of  $32 \times 28$  m and composed of 80 convex objects. The stereolithography model was converted in a porous hexahedral mesh by the application of the GJK algorithm. The computational domain size was  $91 \times 76 \times 19$  m and three different mesh sizes were initially considered: with cells of 0.60 m (215,000 cells); 0.30 m (1,782,246 cells) and 0.20 m (5,987,280 cells). Figure 44 (b) shows the volume porosity obtained for each computational cell by using the mesh size of 0.30 m. This mesh size was chosen based on a previous mesh test.

By considering the porous model presented in Figure 44 (b), a fluid flow of air at was simulated using the developed PFS tool. At the an air entrance at 70 m/s was considered, while the opposite side was set as outflow boundary (specified static pressure and the other variables extrapolated from interior nodes). All the other boundaries of the computational domain were set as solid surface as well as the contours of the geometric

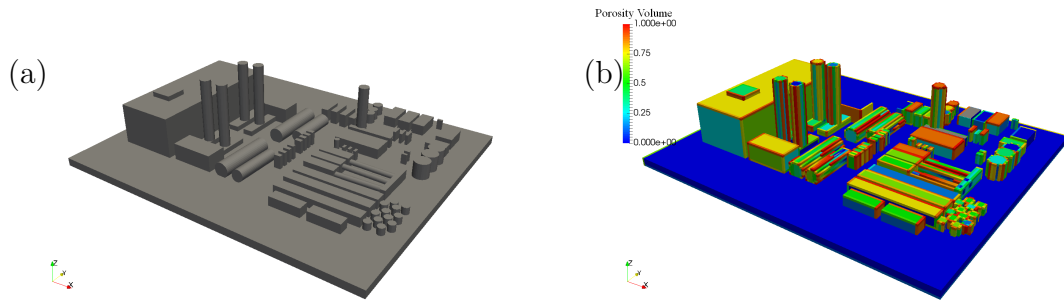


Figure 44 – Idealized offshore module adopted for flow simulation in a complex geometry: in STL format (a) and parametrized geometry as porous media using a mesh of 0.3 m size (b).

objects.

Figure 45 (a) shows the velocity field in the process area of the platform given by PFS simulation. The results were taken in a plane 1.0 m over the platform floor. The streamlines that contorns all objects in the process are are presented in Figure 45 (b).

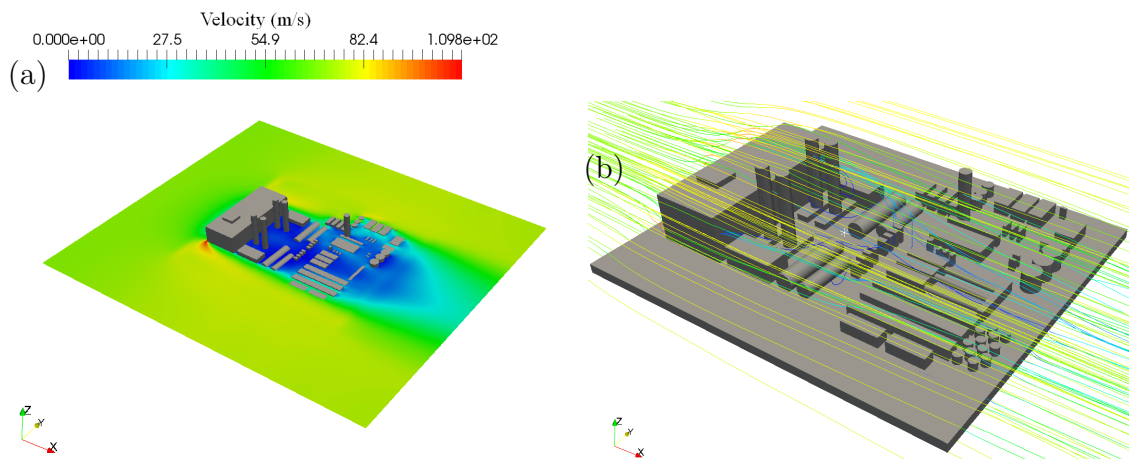


Figure 45 – Ventilation in a ofshore module provided by PFS simulation: scalar velocity field (a) and streamlines (b).

A similar simulation was performed considering the traditional CFD methodology (that solves a fluid mesh) by using the ANSYS-CFX CFD tool. The stl geometric model shown in Figure 44 was adopted and the computational domain size was the same one applied in PFS simulation. A tetrahedral mesh was used with the minimum and maximum mesh size set as 0.075 m and 2.0 m, respectively, which yield 1,400,364 computational cells. The domain boundary conditions applied here were also similar to the ones set in PFS simulation: an entrance of air at 70m/s at the x left side; an outflow condition (static pressure equal to zero) at the x right side and solid surface at the other sides of the computational domain. Another important feature is that the upwind scheme was applied here.

A quantitative analysis was performed considering both PFS and CFX simulations. Velocity profiles were taken in different regions of the process area. The regions were defined by considering 5 monitor lines into the process area. Each line was addressed with 100 monitor points that covers the y direction from the beginning until the end of the computational domain. The lines were positioned based on the origin of the platform deck (floor) in the coordinate x and z equal to (0,0), therefore: Monitor Line 01 - x = 10 m and z = 3 m; Monitor Line 02 - x = 10 m and z = 5 m; Monitor Line 03 - x = 16 m and z = 3 m; Monitor Line 04 - x = 21 m and z = 1 m; Monitor Line 05 - x = 21 m and z = 5 m.

Figure 46 presents the comparison between PFS and CFX simulation results. For all considered regions, the velocity profile given by the PFS is very similar to the profile provided by CFX simulation, so the developed tool is also capable to capture the turbulent flow behaviour in complex geometries in a real scale.

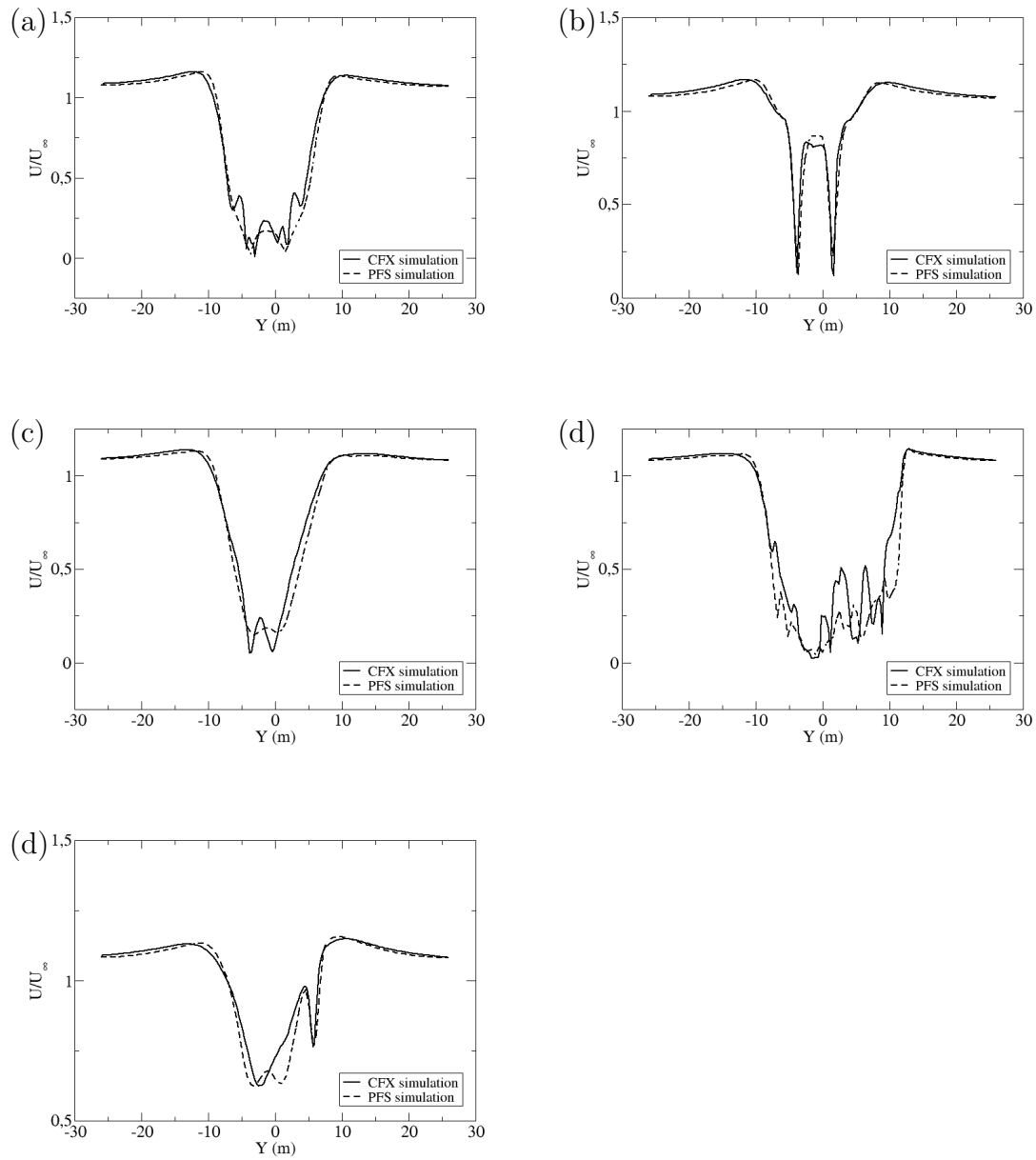


Figure 46 – Comparison of the simulation of air flow in a offshore module provided by the developed PFS tool and ANSYS-CFX. Velocity profiles were taken at five different regions of the process area. Monitor Line 01 (a); Monitor Line 02 (b); Monitor Line 03 (c); Monitor Line 04 (d); Monitor Line 05 (e).

## 6 Case Studies of reacting flows: Gas Explosion

### sion

Once the PFS code was capable to solve three-dimensional turbulent flows by solving the Navier-Stokes equations coupled with the  $k - \varepsilon$  turbulence model, the combustion modelling was implemented into the code in accordance with the modelling described in Section 3.5.

This chapter presents the simulations for reacting flows. Initially, a series of tests was conducted to analyse how the combustion model implemented in the code behaves with the novel approach for the parametrisation of the geometry. The second part comprises the comparison of numerical findings with available experimental data.

### 6.1 Preliminary combustion test

The main features of the combustion process are investigated in this section. The adopted approach follows the parameters of pre-mixed combustion as described in Chapter 2).

Initially, typical reacting flows were simulated in a rectangular chamber with no obstruction or porous region preventing flow. The computational domain was built measuring 0.5 m in length (x direction) and 0.11 m in y and z directions. It was considered that all domain was fully loaded with a flammable mixture of methane and air and the ignition point was placed at a distance equal to 0.1 m from the beginning of the computational domain. Figure 47 shows the schematic representation of this simulation. Table 6 presents the variables for the set up.

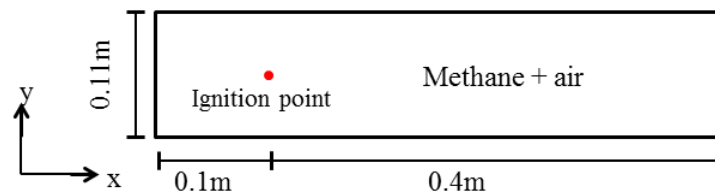


Figure 47 – Schematic representation of a preliminary combustion simulation. The computational domain presents no obstruction and is fully load with a flammable mixture of methane and air.

Following the initial conditions for combustion simulation (Section 4.3.1.1) the reaction progress variable was set to zero in all domain and its value was ramped to a

Table 6 – Main set up features for a preliminary combustion simulation.

Parameter	Condition
Fluid Type	1 - with methane properties
$P_{ref}$ and $T_{ref}$	100 kPa and 300K, respectively
$P_{down}$	99.9 kPa
CFL, Smooth Factor and Residual	0.4; 0.05 and $10^{-4}$

value close to 1.0 at the ignition point. Solid boundary conditions were applied at the beginning of the computational domain (left side on x direction) and the lateral walls. Results of this simulation can be visualised in Figures 48 - 50. Considering the reaction progress variable ranging from 0.0 to 1.0, the value of "c" close to 0.5 means the flame surface region where the combustion reaction takes place. The reaction zone can be easily identified based on the progress variable value.

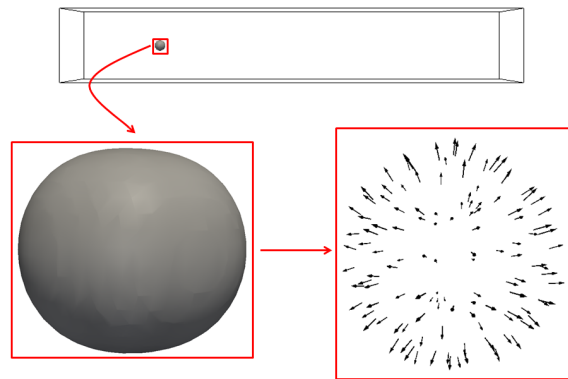


Figure 48 – Three-dimensional flame countour ( $c=0.4$ ) and vector field at 4.6ms after ignition.

Figure 48 shows that the flame presents a spherical shape, which is in accordance with the literature description of a flame surface and also with the combustion model applied in this work (section 3.5). Besides that, the presence of vectors in the three directions of the velocity field indicates that, as expected for an explosion process, the flame propagates in all directions.

Figure 49 shows the reaction progress variable "PV" (a), fuel fraction "FF" (b), pressure (c) and temperature (c) fields. When analysing the reaction progress variable and fuel fraction, in the kernel of the explosion, the reaction progress variable achieves its maximum value while the fuel fraction takes minimum values. Such a behaviour is explained by the fact that when the explosion starts, near the ignition point, the reagent is converted into products, so the variable PV assume values close to a unit and fuel fraction decreases, once the fuel is consumed during combustion reaction. When analysing

the pressure and temperature fields (Figure 49 (c) and (d)) it is observed that both pressure and temperature are increased near the ignition point. Such a behaviour is in accordance with literature that describes the explosion as a process with an increase in volume, pressure and temperature (ECKHOFF, 2005). The literature also indicates that during an explosion process the temperature can achieve values close to 2,000 K, which is very close to the maximum value obtained in this preliminary simulation (1,850K).

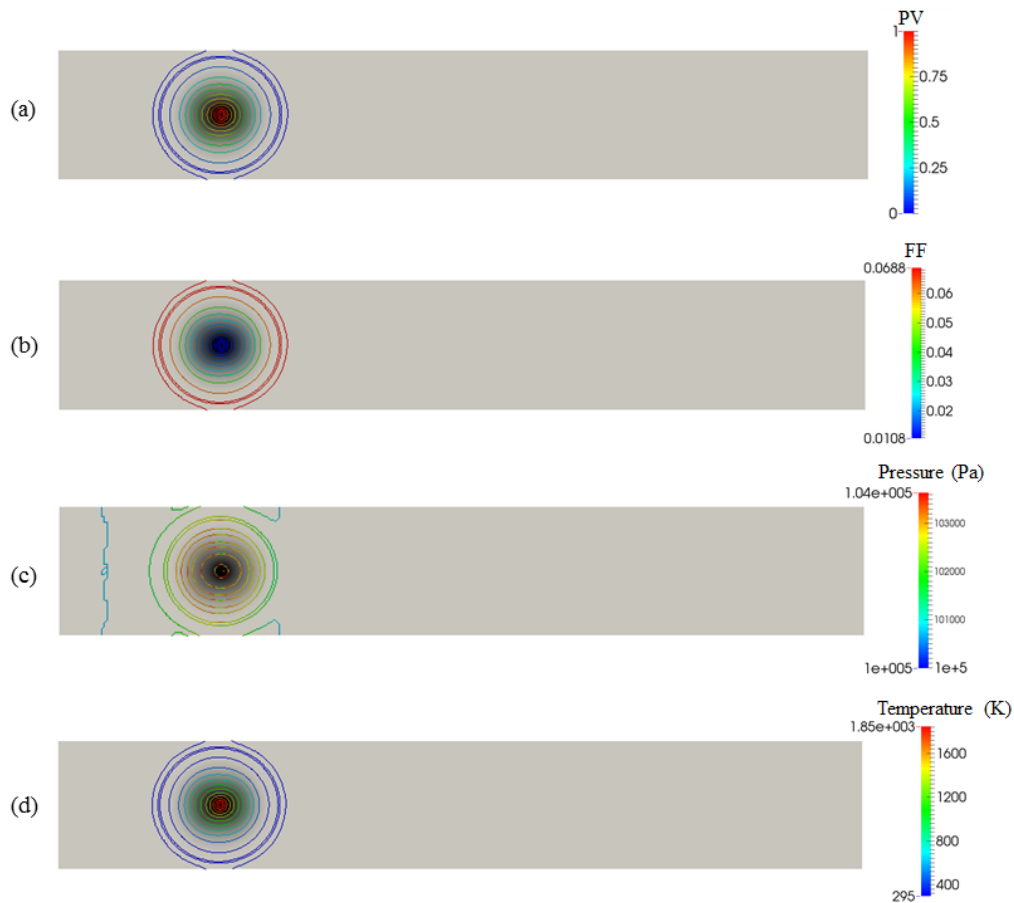


Figure 49 – Preliminary explosion simulation using PFS code: reaction progress variable (a), fuel fraction (b), pressure (c) and temperature field (d) at 4.6ms after ignition.

The last consideration concerns the analysis of the main dimensionless numbers applied to characterise the premixed turbulent flames: Damköler and Karlovitz number. As described in Section 2.1.2.3 and visualised in Figure 2 the flamelet regime is achieved when Karlovitz number is less than or close to a unit. When analysing Figure 50 (b), it is possible to observe that the Karlovitz number assumes values less than 1.0 at the flame region ( $PV \approx 0.5$  in Figure 49 (a)), which ensures that the numerical simulation of an explosion is conducted during a flamelet regime. Concerning Damköler's number, it is mathematically inverse to Karlovitz, so at the same region that represents the flame surface, the maximum values of Damköler are achieved, as observed in Figure 50 (a).

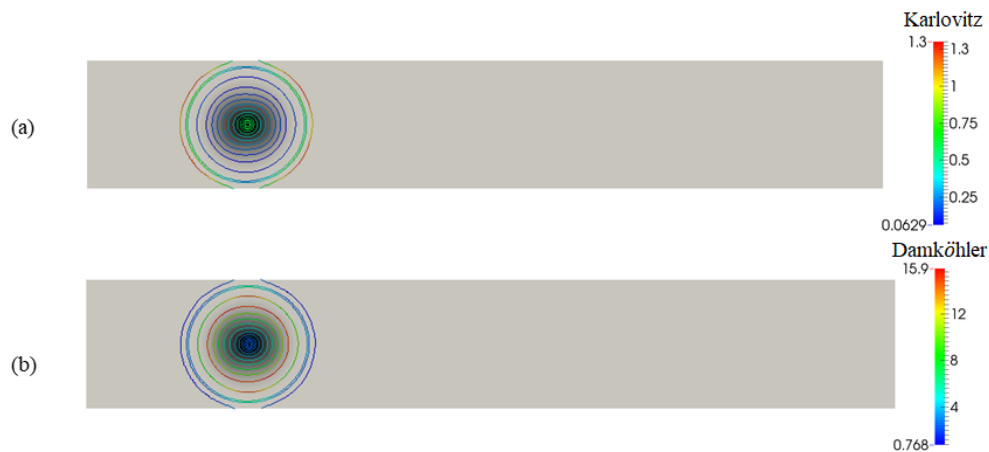


Figure 50 – Preliminary explosion simulation using PFS code: non-dimensional Damköhler (a) and Karlovitz (b) numbers at 4.6ms after ignition.

### 6.1.1 Sensitive tests of combustion model

Sensitive tests were also conducted to understand the explosion behaviour when changing some parameters of the adopted combustion model. The same computational domain presented in Figure 47 was used here, however the ignition point was placed at its beginning. The flame advancement was analysed by considering its position into the domain with time.

- **Influence of the initial laminar modelling**

As explained in Section 3.5.2, even though the overall behaviour of an explosion is turbulent, a laminar phase is present at the beginning of the process. To analyse the influence of this initial laminar phase two simulations were conducted: the first one with all combustion modelled as a 100% turbulent process and the second one with a laminar initial phase. The transition from laminar to the turbulent phase was modelled as suggested by Vianna e Cant (2014) (Equation 3.53). Figure 51 presents the comparison between these two simulations by considering the flame position into the computational domain with time. For the same instant of time, the distance travelled by the flame into the computational domain when modelling a initial laminar phase is smaller than when all process is modelled as turbulent. As explained in Section 2.1.1 the flame speed is enhanced by turbulence mechanisms so when a initial laminar phase is included in combustion modelling, the flame advancement is delayed, the results presented in Figure 51 are in accordance with this behaviour.

- **BML combustion model parameters analysis**



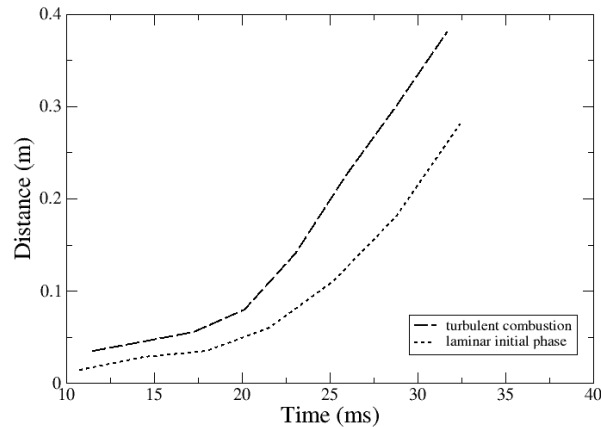


Figure 51 – Comparison between numerical simulations using a turbulent combustion modelling and a combustion model with an initial laminar phase.

The reaction rate modelling provided by the Bray-Moss-Libby (BML) combustion model, used in this work, presents different adjustable parameters, as shown by equations 3.46 - 3.48. The variable  $c_L$  stands up among them since it directly affects the reaction rate. Bray *et al.* (1985) pointed out that the variable  $c_L$  must be increased to bring down the reaction rate. In this part of the study, a sensitive test was conducted to evaluate the response of developed solver for explosion simulations when different values are applied to the variable  $c_L$ : 1.0; 2.5 and 5.0. The result of such analysis is shown in Figure 52.

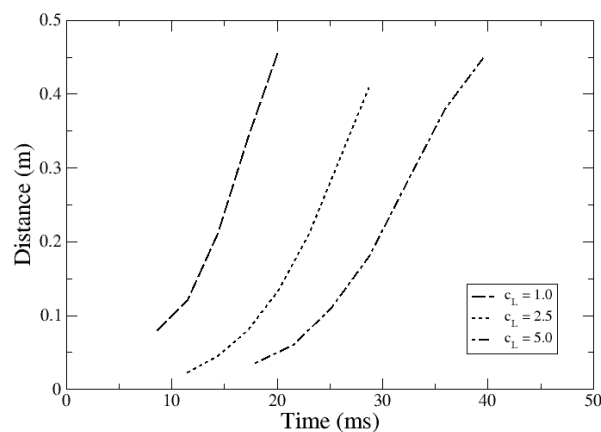


Figure 52 – Analysis of the influence of adjustable variable  $c_L$  of BML reaction rate model on flame propagation.

When increasing the value of  $c_L$ , the flame advance into the chamber becomes slow, which indicates that this variable affects the flame propagation velocity. It also

confirms the influence of  $c_L$  on reaction rate when the BML combustion model is used. As expected the reaction rate is decreasing when  $c_L$  take great values.

## 6.2 Case studies of combustion flows

### 6.2.1 Study of a turbulent premixed flame past over an obstacle

To analyse the explosion behaviour when a turbulent premixed flame interacts with a parametrised geometry, a combustion chamber obstructed by a single obstacle was initially considered Gubba *et al.* (2011). Following the PFS methodology to define the geometry under study and the computational domain, a squared box with 12 mm of size was placed into a computational domain with  $50 \times 50 \times 250$  mm of dimensions, which yield a blockage ratio of 25%. The squared obstacle was located at a distance equal to 100 mm from the chamber basis. Figure 53 presents a schematic representation of the this situation.

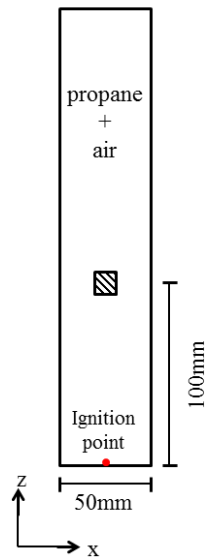


Figure 53 – Schematic representation of a combustion chamber obstructed by a single obstacle.

The lateral sides and the chamber bottom were fully closed, so wall boundary conditions were set up at these regions, while the top of the chamber was opened. The chamber was loaded with a stoichiometric mixture of propane and air and the ignition point was placed at its bottom, as shown in Figure 53. Computational cells with 1.2 mm of length were used to build the numerical grid and the simulation parameters were the same presented in Table 6. A value of variable  $c_L$  equal to 2.0 was applied.

The flame position and overpressure time series are presented in Figure 54. The numerical results provided by PFS simulation were plotted against experimental data.

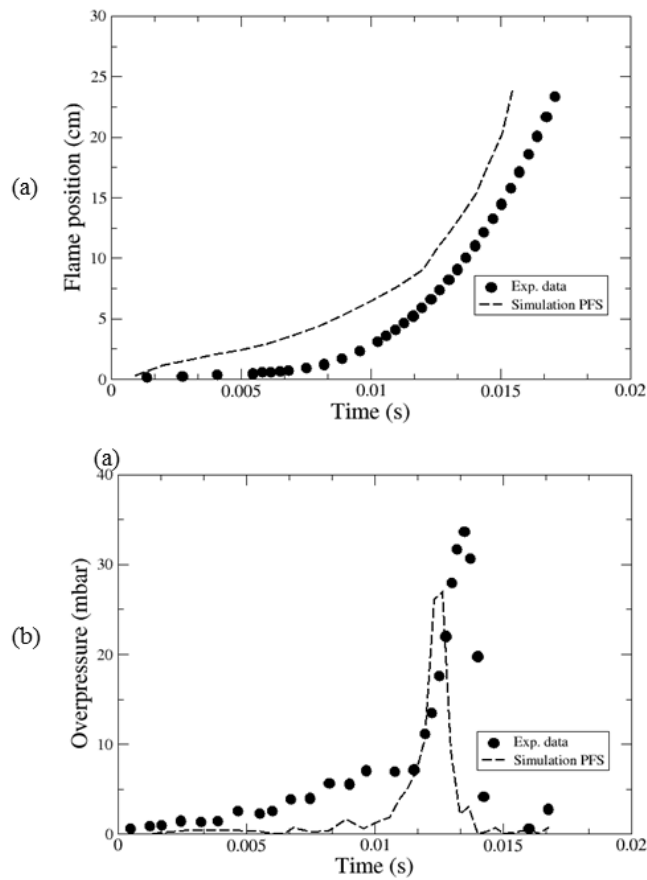


Figure 54 – Time histories of flame position and overpressure of a combustion chamber obstructed by a single obstacle.

Figure 54 (a) shows that the flame goes through the chamber relatively slowly until it reaches the obstacle (about 12 ms after ignition starts) for both experimental and numerical findings, although the PFS RANS predicts earlier flame propagation. When considering the overpressure generated during the explosion (Figure 54 (b)), numerical results also present a good agreement with experimental data. Experimental test shows that the overpressure peak of 34 mbar peak occurs about 13.5 ms after ignition. PFS simulation predicts an overpressure peak of 28 mbar at 12.3 ms which is of about 17% lower than the value given by the experimental test.

Figure 55 presents the flame position into combustion chamber at different time instants after ignition starts. The flame locations were obtained through the calculated reaction rate progressive variable "c". The series of images is compared with experimental data Gubba *et al.* (2011). The PFS RANS simulation is capable to reproduce the turbulent flame structure and the flame propagation.

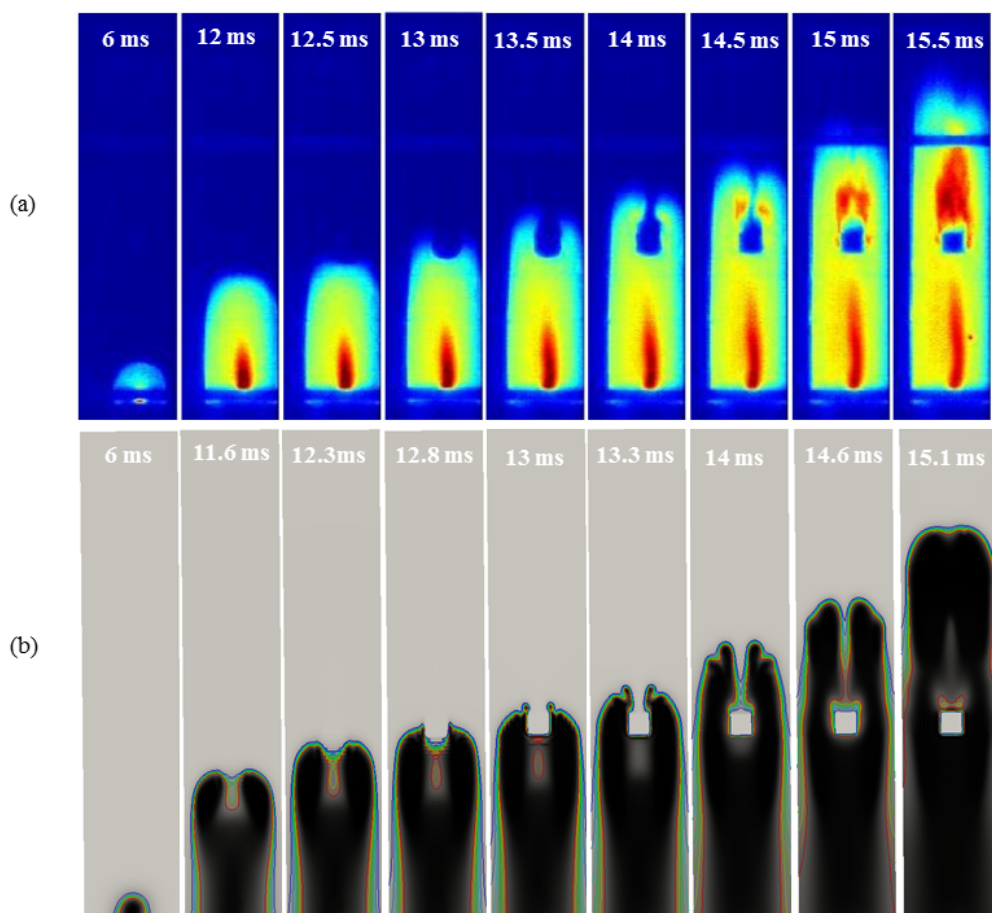


Figure 55 – Flame position into a combustion chamber obstructed by a single obstacle at different time instants after ignition starts. Experimental results are observed in (a) while simulation results provided by the developed solver are presented in (b)

### 6.2.2 Analysis of a turbulent premixed flame accelerated by a sequence of obstacles

The second study of an explosion process was based on Patel *et al.* (2002) and Sarli *et al.* (2009) work, who have studied combustion in a chamber with obstacles. The chamber comprised a rectangular box with size of  $150 \times 150 \times 500$  mm. Three obstacles ( $75 \times 150 \times 12$  mm) were positioned at 100 mm spacings within the chamber. The bottom end of the chamber was fully closed while the upper end was open. The chamber was filled with a flammable mixture of methane and air and the ignition point was placed on its base. Figure 56 presents the schematic representation of this study.

The simulations were performed considering the bottom end and the lateral sides of the chamber as wall (with zero fluxes of any quantity), the upper end was set as open boundary (pressure gradient equal to zero) and the obstacles were described as porous region. The simulation parameters are in Table 6 and the adjustable parameter for

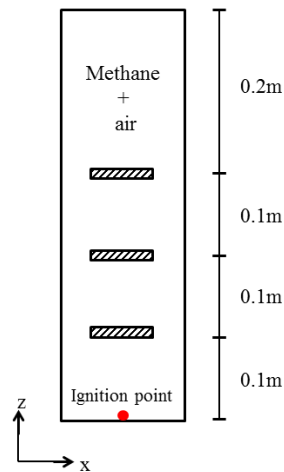


Figure 56 – Schematic representation of a combustion chamber with three obstacles positioned at 100 mm spacings within the chamber.

the combustion model  $c_L$  was taken as 3.0. Figure 57 shows the flame structure within the combustion chamber at different time-steps. The flame surface in numerical images was obtained by considering the reaction progress variable ( $c$ ) equal to 0.4. Numerical results are compared with experimental data provided by Patel *et al.* (2002) .

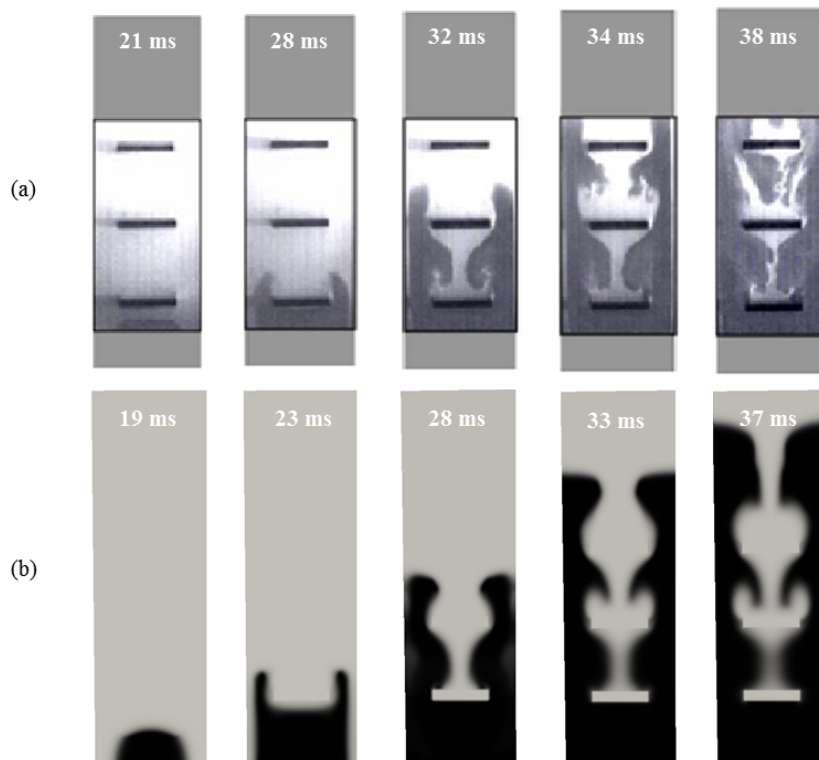


Figure 57 – Time sequence of the flame position at the central plane of the chamber: experimental data (PATEL *et al.*, 2002) (a) and numerical simulation (b).

The present RANS simulation predicts the features of the flame propagation.

As observed in the experiment (Figure 57 (a)), the flame front achieves the first obstacle with an incomplete consumption of the fuel mixture in the upstream region of the chamber and then it is separated into two opposite flame fronts. The flame advances through downstream the object as a jet behaviour and tends to curl at the centreline of the chamber. The same sequential behaviour is verified for the following two obstacles. In the numerical simulation, the flame front impinging onto the first obstacle at the time equal to 23 ms, while experimental data indicates that such interaction occurs 28 ms after ignition. Such a difference between numerical finds and experimental observations for the flame position with time tends to reduce as the flame goes through the chamber. Figure 58 shows the comparison between numerical results and experimental data (PATEL *et al.*, 2002) for time evolution of the flame location inside the chamber. The flame location was obtained by considering the maximum axial distance of the flame front from the ignition point.

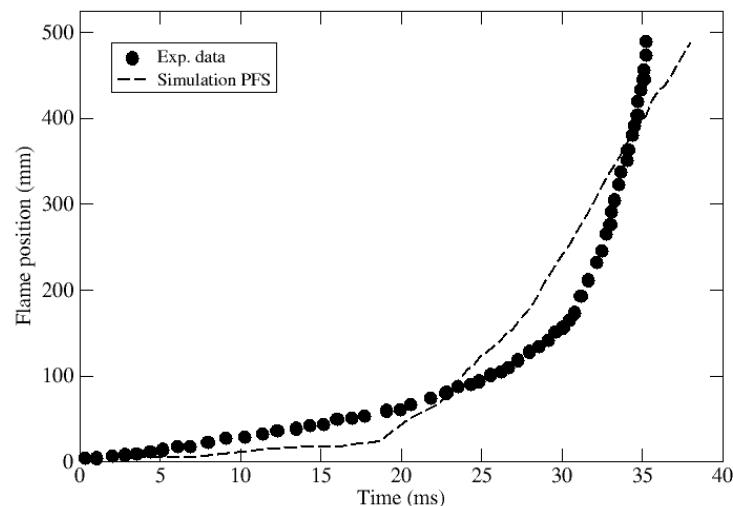


Figure 58 – Comparison between numerical simulation and experimental data for the flame location inside the chamber.

Figure 58 shows that, at the first 20 milliseconds after ignition, the flame propagates slowly into the chamber and the numerical findings agree with experimental data. In the second propagation stage, the flame tip reaches the first obstacle and as observed in Figure 57 the numerical simulation early predicts this arrival (within almost 23 seconds). In the last propagation phase, the flame interacts with obstacles and it is accelerated. A slop is observed on the curves. Here, it is verified that the numerical simulation provides a slower flame propagation compared with experimental data.

The flame speed profile provided by numerical results was also compared with experimental data and is presented in Figure 59. The present simulation is able to capture

the overall flame speed behaviour presented in the experimental test although, quantitatively, there is a lack of agreement between the trend curves.

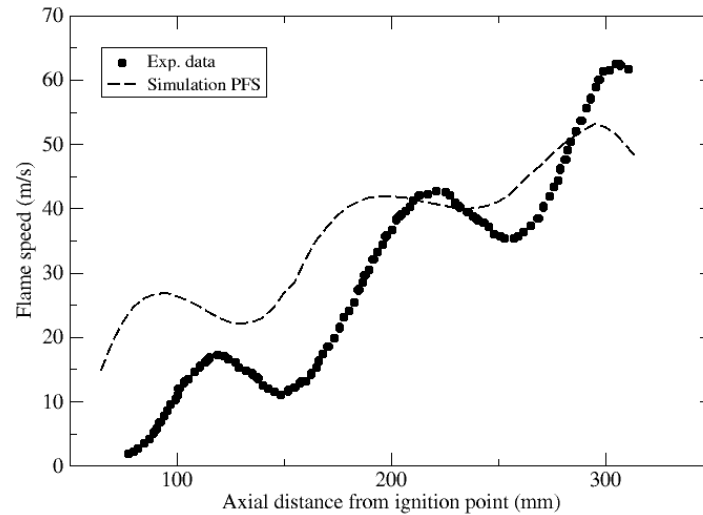


Figure 59 – Experimental data and corresponding numerical predictions for the flame speed profile along the axial distance from the ignition point.

As described for turbulent combustion process, the reaction rate is increased by turbulence effects, which is enhanced by the presence of obstacles in a fluid flow. Thus, when the flame front reaches an obstacle, its velocity is increased and then tends to decrease. This trend is repeated in each chamber obstacle as visualized in Figure 59. Numerical simulation provides greater values of flame speed than the experimental data at the beginning of ignition. The intensity tends to decrease as the flame moves through the other obstacles.

The analysis confirms the flame advancement trend discussed above, wherein the flame tip interacts with the first chamber obstacle earlier on the numerical simulations than in the experimental observations. For both numerical and experimental results, the flame arrives at last obstacles at similar time. This behaviour can be explained by the fact that the porosity distributed model applied in this work to describe the geometrical obstacles may not produce enough turbulent kinetic energy, so the turbulence effect on the flame speed is not observed in numerical findings as well as in experimental data.

The last quantitative analysis of this study comprises the verification of the overpressure generated during the explosion. Figure 60 shows the pressure time history at the bottom end of the chamber for both RANS simulation and experimental data. Numerical data were obtained by considering the mean pressure field in a defined region at the end of chamber domain. It is possible to observe that the numerical model predicts

the dominant peak pressure as observed in experimental test. The maximum overpressure was 27% lower than experimental data, which can be due by the presence of a sealing membrane that was not considered in numerical simulations. However, for both experimental and numerical results, the pressure peak was found at around 38 ms after ignition starts which occurs when the flame tip reaches the third obstacle.

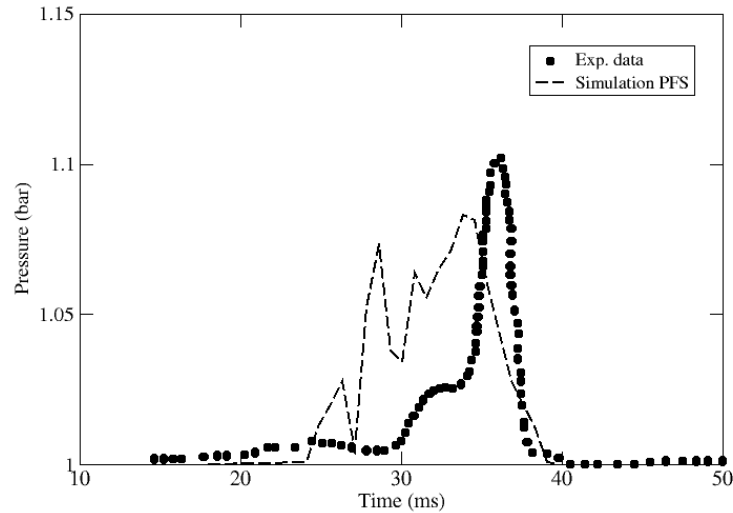


Figure 60 – Time pressure history at the bottom end of the chamber - comparison between numerical results and experimental data.

Figure 61 shows the dimensionless *Damköler* and Karlovitz numbers at the flame front region. In the presented images, the contour lines were obtained for values of reaction progress variable "c" ranging from 0.2 to 1.0. It is possible to observe that the values of  $Ka$  at the flame front are close to unity, which is in accordance with the Bourghi diagram (Figure 2) when flamelet modelling is applied to combustion problems.



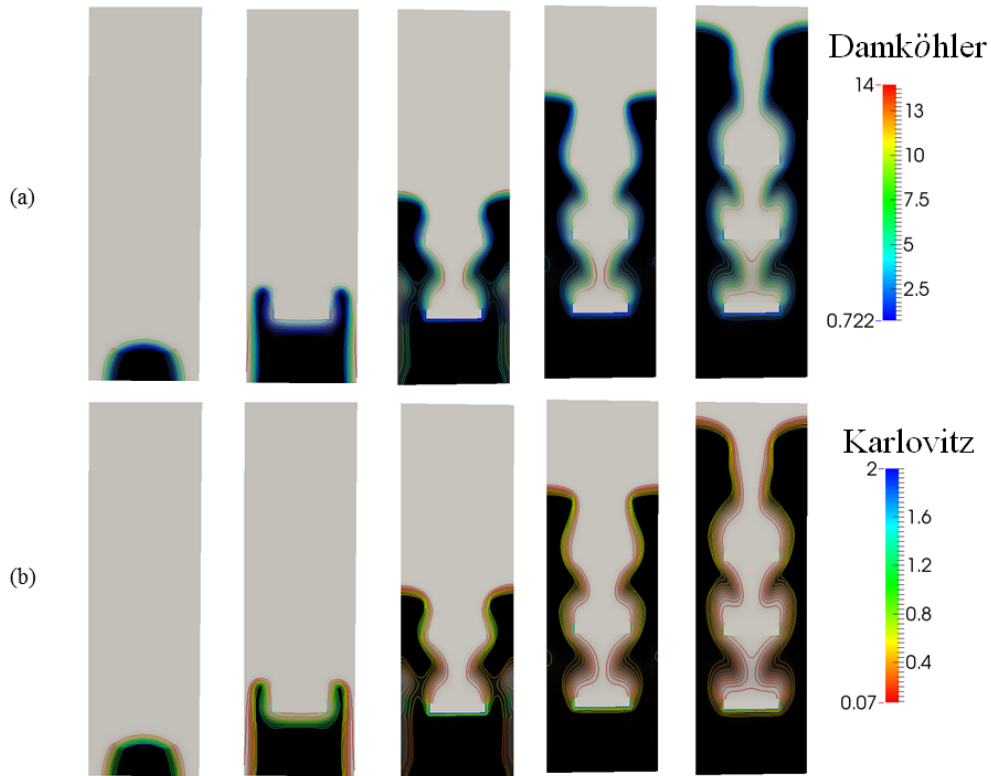


Figure 61 – Dimensionless *Damköhler* and Karlovitz numbers at the flame front region for different time instants.

### 6.2.3 Study of turbulent combustion in a chamber with different obstacles blockage ratio (OBR)

The combustion process in a chamber with obstacles at different blockage ratios (BR) reported by Chen *et al.* (2016) was also analysed by numerical simulation using the developed code. The chamber consists of a rectangular vessel with dimensions of  $500 \times 80 \times 110$  mm. The left side (the beginning of chamber) was considered fully closed and was used as basis to the ignition point while the right end was set up as opening vent. A rectangular obstacle 40mm thick was placed at about 210 mm from ignition point. The **blockage ratio** (BR) is related to the gap between the obstacle size  $H$  and the channel height  $h$ , so:

$$BR = 1 - \frac{H}{h} \quad (6.1)$$

Following the relation given in Equation 6.1, three obstacles blockage ratio were analysed by varying the obstacle height, so  $BR = 0.3$ ;  $0.5$  and  $0.7$ . Figure 62 presents the three different chamber configurations considered in this study.

As shown in Figure 62, the combustion chamber was considered fully loaded with a stoichiometric mixture of methane/air. The boundary and initial conditions and

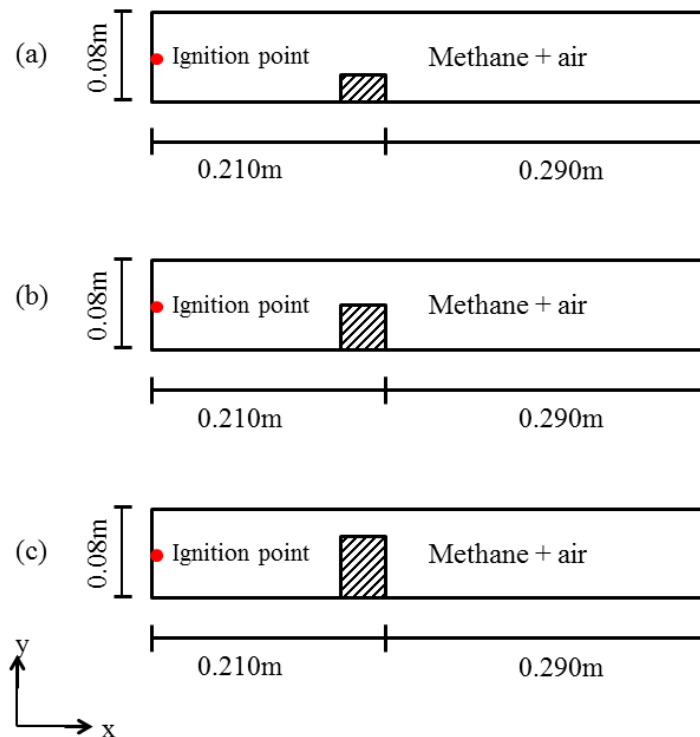


Figure 62 – Schematic representation of a combustion chamber with three different obstacles blockage ratios:  $BR = 0.3$  (a);  $BR = 0.5$  (b) and  $BR = 0.7$  (c).

simulation parameters were the same used before (section 6.2.2). A mesh grid with 4mm of length was used and the variable  $c_L$  was set up with a value equal to 5.0. Figures 63 to 65 presents the flame propagation into the chamber obstructed by obstacle with different blockage ratios. The numerical results given by simulation using the developed PFS code are compared with experimental data provided by Chen *et al.* (2016) when the chamber blockage ratio was 0.3 (Figure 63), 0.5 (Figure 64) and 0.7 (Figure 65).

Analysis of the sequence of flame propagation shown in Figures 63 to 65 allows the identification of four main stages for the flame propagation. The first stage occurs just after ignition until  $t=10$  ms and the flame presents a spherical shape with laminar propagation behaviour. At the second stage, the flame goes through the chamber and assumes an extended-shape with a greatly increase on its surface area. A third stage occurs when the flame reaches the obstacle and assumes a jet behaviour by passing through a throttled channel. The last stage is characterized by the presence of large recirculation zones behind the obstacle and the vortex interaction with flame. For all considered blockage ratio, numerical results can predicts the flame shape and also its propagation stages. However, as observed in previous numerical simulation of turbulent combustion flows, the PFS simulation tends to predicts an early interaction between flame front and obstacle.

Figure 66 shows the numerical vector field obtained by the present RANS

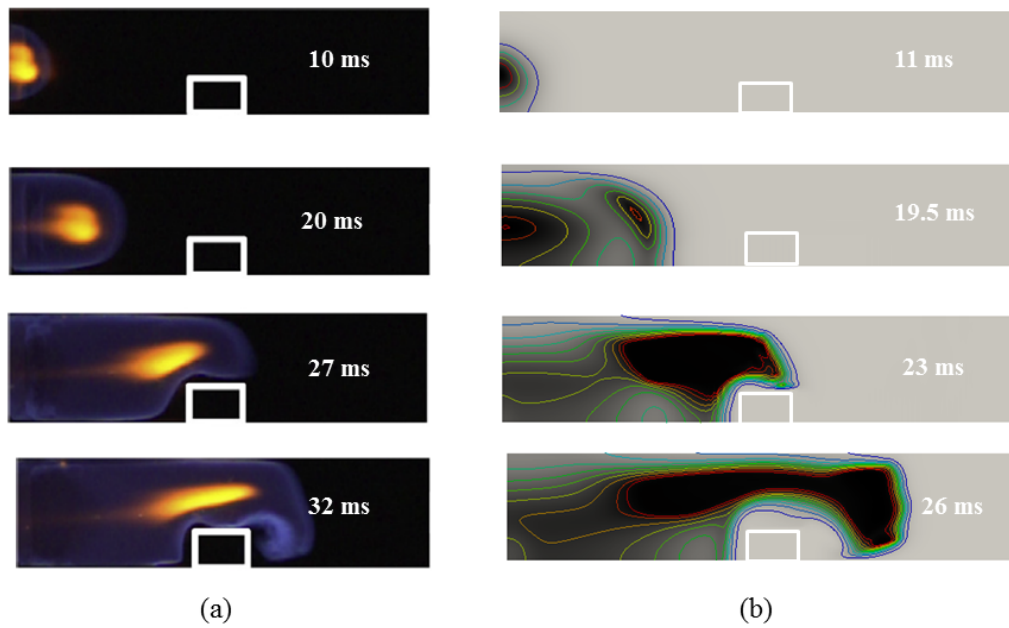


Figure 63 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with  $BR = 0.3$ : comparison between experimental data (CHEN *et al.*, 2016) and the PFS RANS combustion simulation.

simulation in a chamber with blockage ratio equal to 0.5. Recirculation zones appears behind the obstacle accompanied by the flame transition. The vortex starts to appears at  $t = 19$  ms as a consequence of the expansion of the combustion products which forces the unburned mixture to accelerate through the chamber. Following the time sequence, the vortex size grows with the increase of the flame speed. At the time instant equal to 25 ms, the flame tip starts to interact with the vortex.

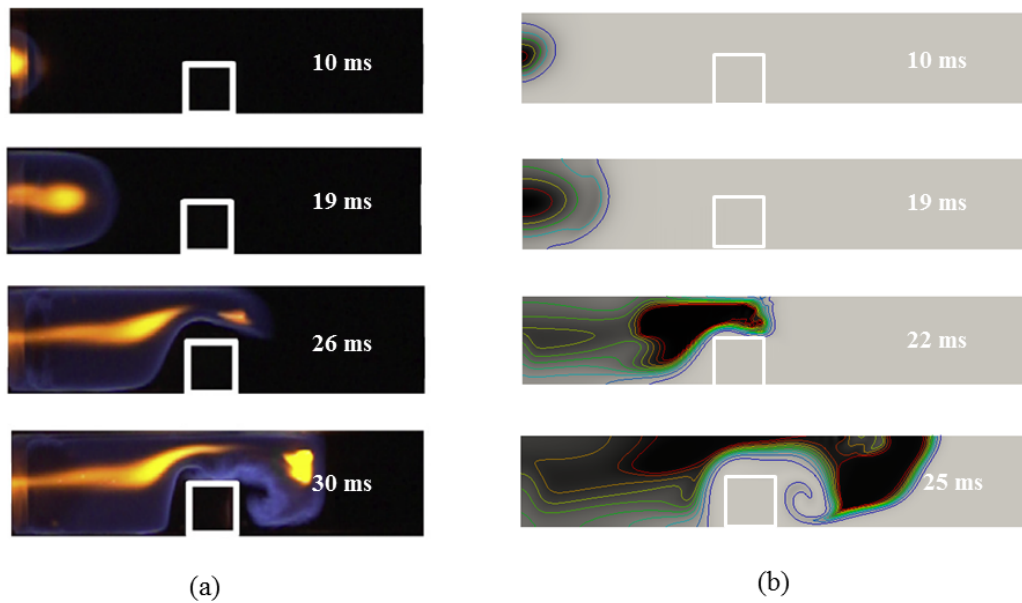


Figure 64 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with  $BR = 0.5$ : comparison between experimental data (CHEN *et al.*, 2016) and the PFS RANS combustion simulation.

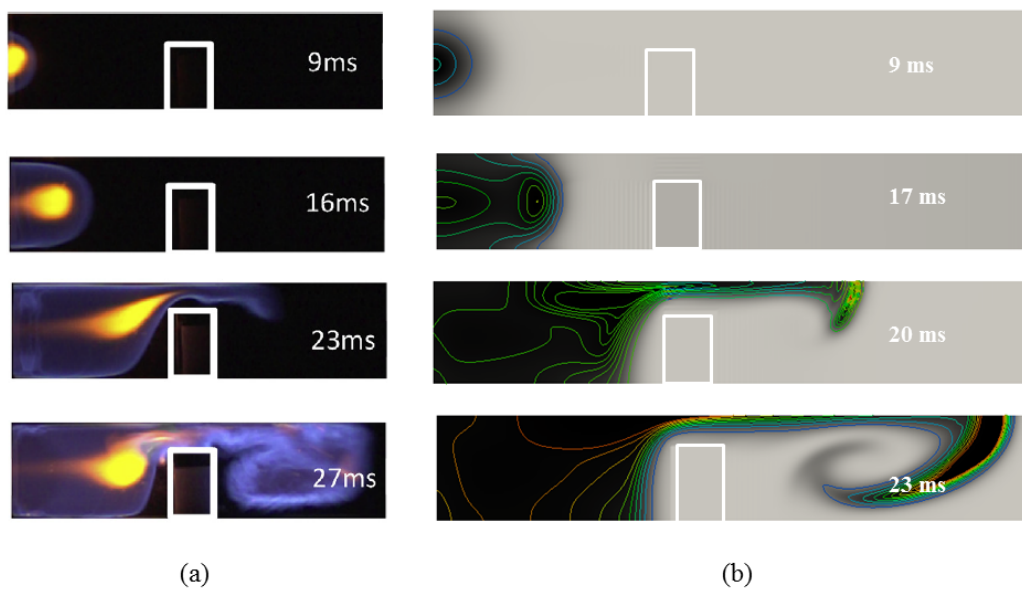


Figure 65 – Premixed air/methane flame propagation in a combustion chamber obstructed by an obstacles with  $BR = 0.7$ : comparison between experimental data (CHEN *et al.*, 2016) and the PFS RANS combustion simulation.

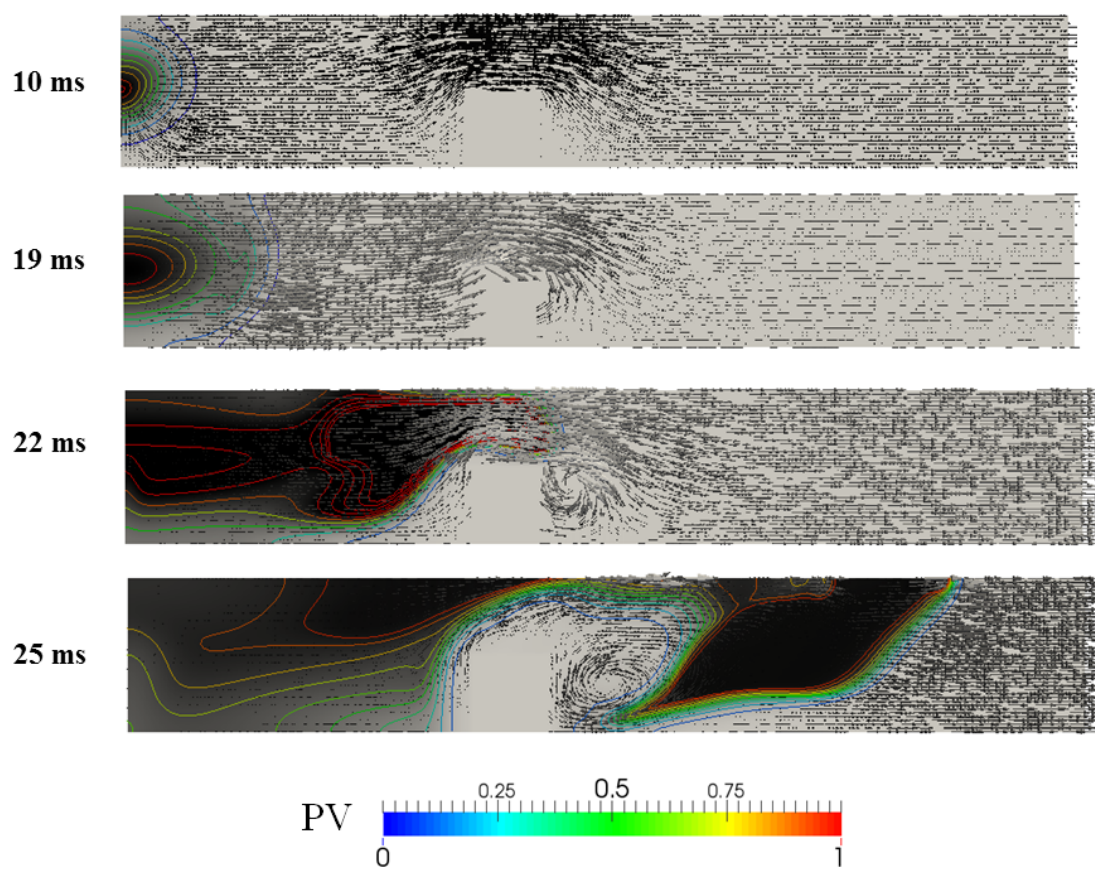


Figure 66 – Vector velocity field generated by the obstacle for the blockage ratio equal to 0.5.

## 7 Conclusions and Future Work

### 7.1 Conclusions

The combination of Gilbert - Johnson - Keerti distance algorithm and the finite volume method was proposed for parametrisation of complex geometries. Regions of the computational domain where obstacles were located, were assigned zero porosity while free zones of the computational domain were assigned unity porosity. Any other region of the computational domain was treated as intermediate porosity following the procedure that checks for collisions between the computational mesh and each primitive solid of the geometrical model. The developed method was applied to describe conventional CAD geometrical models.

Initially, a three-dimensional Euler solver was developed based on the initial two-dimensional Euler solver. Flows over a bump were investigated and it was verified the flux calculations over the boundaries and computational cells. The numerical findings in this stage were consistent to the literature description for the solution of compressible flows.

In the next step, the developed three-dimensional Euler solver was applied to calculate flow fields using parametrised geometries by the GJK algorithm. Flows over a buff body were analyzed and it was verified that the flow moves away from parametrised geometries in a similar manner to the conventional geometries, which indicates that the proposed parametrisation method can be used to represent geometries in fluid flow studies.

Some tests were conducted to verify the code implementation. The main simulation parameters (mesh size, smooth factor and CFL number) were investigated. Their influence in the numerical convergence and in the flow solution proved to be consistent with what is expected when considering the solver features. A comparison was also performed between the developed solver and an established commercial CFD code.

The modelling approach used at the  $k - \varepsilon$  turbulence model was applied in the third part of this work to account for the viscous effects and extra turbulence generation in parametrised geometries. Dedicated turbulent flows over buff bodies were investigated. The numerical findings were in accordance with experimental data for the velocity field.

As far as the turbulent kinetic energy and the fluctuating velocity go, the approach has been able to reproduce the overall trend. However, numerical findings tend to over-predict the experimental results for flow over a step. For the flow over cylindrical obstacles the results are under predicted when compared with experimental data.

Comparison of the turbulent features of the current work with other numerical results where RANS approach and  $k - \epsilon$  turbulence model has been applied show similar behaviour. Considering that the velocity field is in accordance with the current understanding of fluid flow and that the numerical results agree with the experimental data, it is likely that the lack of agreement between the numerical results for the fluctuating velocity and experimental data is due to limitations of the  $k - \epsilon$  formulation. The presence of circulating flow at the wake of obstacles poses an additional challenge when two-equations models concerning the turbulent kinetic energy and the rate of dissipation of the turbulent kinetic energy are considered.

Regarding the simulation of reacting flows, the analysis of three different cases of turbulent combustion process has shown that the developed method is capable of describing the flame structure and its propagation behaviour within the framework of traditional laminar flamelet model.

As expected for the turbulent combustion process, overpressure peaks were captured by the numerical simulations. The results calculated by the developed solver tends to underpredict the maximum pressure values ( $\approx 20\%$ ) achieved in experimental tests.

A slight lack of agreement was observed between numerical findings and experimental data when analysing the flame speed. The presented method tends to predict a faster flame propagation for unobstructed flows and a slower propagation when the flame interacts with an obstacle. These trends may indicate the necessity of improvement on turbulence model, once the combustion reaction rate mainly depends on the turbulent field.

The Damkhöler hypothesis for reacting flows were verified under the influence of GJK parametrisation. For all considering combustion simulations, the Damkhöler (Da) number was greater than a unity, which indicates that the chemistry is very fast compared to fluid mechanical scales. Therefore, the combustion process occurs in the laminar flamelet regime, assumed as basis for the combustion modelling in this work.

## 7.2 Future Work

The combination of the finite volume method and the Gilbert - Johnson - Keerti distance algorithm for parametrisation of geometries in numerical simulations provides satisfactory results. Findings have shown that a promising approach has emerged, however, further work is necessary to improve the developed solver.

Implementation of a high order numerical method, such as *eno* or *weno* scheme, can be used to provide a better solution of the governing equations. The use of a high

order method will replace the application of artificial viscosity, an adjustable factor that reduces the oscillations generated by the central difference scheme.

Besides that, a better turbulence approach is required to improve the predictions of turbulence generated by the porous mesh.

The reaction rate in the BML combustion model (BRAY *et al.*, 1985), used in this work, depends on the empirical correlation  $f\left(\frac{u'}{u_L}\right)$  (Equation 3.48) to compute the integral length scale of the flame wrinkling. Therefore, as in any empirical function, different adjustable parameters are applied here. As future work, it is suggested the replacement of such an empirical correlation to reduce the necessity of adjustable parameters in the BML combustion model. A possible solution was already proposed by Matos (2014), who pointed out that the integral length scale of wrinkling can be computed based on the turbulence dissipation rate ( $\varepsilon$ ). The model was successfully implemented in a spreadsheet for explosion calculations in a large-scale chamber with a vent.

The developed approach works well for all selected cases of non-reacting and reacting turbulent flows. However, in this research, only simple and small scales geometries were deeper investigated. It is proposed to evaluate the solver performance for a large range of length scales, as in a real process plant. Another suggestion comprises the analysis of different cases of turbulent reacting flows, since only the combustion process was addressed here.

At least, it would be interesting to compare the numerical findings of the developed solver with an establish CFD tool for gas explosion, as FLACS, for example. Since FLACS uses the Porosity Distributed Resistance model to represent a geometry under study, the comparison could provide a consistent analysis of the implementation of turbulence and combustion models considered in this work.



# Bibliography

ABU-ORF, G. *Laminar flamelet reaction rate modelling for spark-ignition engines*. Tese (Doutorado), 1996.

ARNTZEN, B. J. *Modelling of turbulence and combustion for simulation of gas explosions in complex geometries*. Tese (Doutorado), 1998.

BERG, A. Van den; MERCX, W.; MOUILLEAU, Y.; HAYHURST, C. Autoreagas-a cfd-tool for gaz explosion hazard analysis. In: *International Conference and Exhibition Offshore Structural Design. Hazards safety and engineering*. [S.l.: s.n.], 1994.

BERGEN, G. van den. *A Fast and Robust GJK Implementation for Collision Detection of Convex Objects*. 1999.

BIRKBY, P.; CANT, R.; SAVILL, A. The application of a laminar flamelet model to confined explosion hazards. *Flow, turbulence and combustion*, Springer, v. 63, n. 1–4, p. 361, 2000.

BJERKETVEDT, D.; BAKKE, J. R.; WINGERDEN, K. V. Gas explosion handbook. *Journal of hazardous materials*, Elsevier, v. 52, n. 1, p. 1–150, 1997.

BRAY, K.; LIBBY, P. A.; MOSS, J. Unified modeling approach for premixed turbulent combustion—part i: General formulation. *Combustion and flame*, Elsevier, v. 61, n. 1, p. 87–102, 1985.

CANT, R.; BRAY, K. A theoretical model of premixed turbulent combustion in closed vessels. *Combustion and flame*, Elsevier, v. 76, n. 3-4, p. 243–263, 1989.

CANT, R.; MASTORAKOS, E. *An introduction to turbulent reacting flows*. [S.l.]: Imperial College Press, 2008.

CANT, S. Modelling and simulation of accidental combustion in complex geometries. In: *5th International Seminar on Fire and Explosion Hazards, Edinburgh*. [S.l.: s.n.], 2007. p. 57–76.

CHEN, P.; LI, Y.; HUANG, F.; GUO, S.; LIU, X. Experimental and les investigation of premixed methane/air flame propagating in a chamber for three obstacle br configurations. *Journal of Loss Prevention in the Process Industries*, Elsevier, v. 41, p. 48–54, 2016.

DISKIN, B.; THOMAS, J. L.; NIELSEN, E. J.; NISHIKAWA, H.; WHITE, J. A. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: viscous fluxes. *AIAA journal*, v. 48, n. 7, p. 1326–1338, 2010.

DURAO, D.; HEITOR, M.; PEREIRA, J. Measurements of turbulent and periodic flows around a square cross-section cylinder. *Experiments in Fluids*, Springer, v. 6, n. 5, p. 298–304, 1988.

ECKHOFF, H. *Explosion Hazards in the Process Industries*. Huston: Gulf Publishing Company, 2005.

- ERICSON, C. *Collision Detection in Interaction 3D Environments*. 2004.
- FAVRE, A. Statistical equations of turbulent gases. *Problems of hydrodynamics and continuum mechanics*, SIAM Philadelphia, PA, p. 231–266, 1969.
- FOTHERGILL, C.; CHYNOWETH, S.; ROBERTS, P.; PACKWOOD, A. Evaluation of a cfd porous model for calculating ventilation in explosion hazard assessments. *Journal of Loss Prevention in the Process Industries*, Elsevier, v. 16, n. 4, p. 341–347, 2003.
- GILBERT, E.; FOO, C. Computing the distance between smooth objects in three dimensional space. In: *Robotics and Automation, 1989. Proceedings., 1989 IEEE International Conference on*. [S.l.: s.n.], 1989. v. 1, p. 158–163.
- GILBERT, E.; JOHNSON, D.; KEERTHI, S. A fast procedure for computing the distance between complex objects in three-dimensional space. *Robotics and Automation, IEEE Journal of*, v. 4, n. 2, p. 193–203, Apr 1988.
- GONÇALVES, H. L.; VIANNA, S. Desenvolvimento de um código euler para escoamento compressível em regimes sônico e supersônico. *Blucher Chemical Engineering Proceedings*, v. 1, n. 1, p. 473–477, 2014.
- GREGORI NERI VOLPATO, R. M. R. M. M. H.; SILVA, M. V. G. da. Slicing triangle meshes: An asymptotically optimal algorithm. *Proceedings of 14th Inter. Conference on Computational Science and Its Applications*, 2014.
- GUBBA, S. R.; IBRAHIM, S. S.; MALALASEKERA, W.; MASRI, A. R. Measurements and les calculations of turbulent premixed flame propagation past repeated obstacles. *Combustion and Flame*, Elsevier, v. 158, n. 12, p. 2465–2481, 2011.
- GUO, Y.; HE, L.; WANG, D.; LIU, S. Numerical investigation of surface conduit parallel gas pipeline explosive based on the tnt equivalent weight method. *Journal of Loss Prevention in the Process Industries*, Elsevier, v. 44, p. 360–368, 2016.
- HEJRANFAR, K.; AZAMPOUR, M.-H. Assessment of cell-centered and cell-vertex finite volume approaches for computation of 2d structural dynamics on arbitrary quadrilateral grids. *CMES: Computer Modeling in Engineering & Sciences*, v. 106, n. 6, p. 395–439, 2015.
- HIRSCH, C. *Numerical computation of internal and external flows: The fundamentals of computational fluid dynamics*. Oxford: Jhon Wiley & Sons, 2007. v. 1.
- HJERTAGER, B. H. Simulation of gas explosions. *Modeling, identification and control, Norsk Forening for Automatisering (NFA)*, v. 10, n. 4, p. 227, 1989.
- HJERTAGER, B. H.; SOLBERG, T.; NYMOEN, K. O. Computer modelling of gas explosion propagation in offshore modules. *Journal of loss prevention in the process industries*, Elsevier, v. 5, n. 3, p. 165–174, 1992.
- JAMESON, A.; BAKER, T. Improvements to the aircraft euler method. In: *25th AIAA Aerospace Sciences Meeting*. [S.l.: s.n.], 1987. p. 452.
- KIM, J.; KLINE, S.; JOHNSTON, J. Investigation of a reattaching turbulent shear layer: flow over a backward-facing step. *Journal of Fluids Engineering*, American Society of Mechanical Engineers, v. 102, n. 3, p. 302–308, 1980.

- KUHL, A.; FERGUSON, R.; CHIEN, K.; COLLINS, J.; OPPENHEIM, A. *Gasdynamic model of turbulent combustion in an explosion*. [S.l.], 1994.
- KUNDU, P.; COHEN, I. *Fluid Mechanics*. San Diego, USA: Academic Press, 2002.
- LAM, C.; BREMHORST, K. A modified form of the  $k-\varepsilon$  model for predicting wall turbulence. *Journal of fluids engineering*, American Society of Mechanical Engineers, v. 103, n. 3, p. 456–460, 1981.
- LAUNDER, B. E.; SPALDING, D. B. The numerical computation of turbulent flows. In: *Numerical Prediction of Flow, Heat Transfer, Turbulence and Combustion*. [S.l.]: Elsevier, 1983. p. 96–116.
- LEA, C.; LEDIN, H. *A review of the state-of-the-art in gas explosion modelling*. [S.l.]: Health and Safety Laboratory Buxton, UK, 2002.
- LYN, D.; EINAV, S.; RODI, W.; PARK, J.-H. A laser-doppler velocimetry study of ensemble-averaged characteristics of the turbulent near wake of a square cylinder. *Journal of Fluid Mechanics*, Cambridge University Press, v. 304, p. 285–319, 1995.
- MALISKA, C. R. *Transferência de Calor e Mecânica dos Fluidos Computacional*. Rio de Janeiro, BR: Livros Técnicos e Científicos Editora S.A, 2004.
- MATOS, R. *Desenvolvimento de um modelo para estimativa da máxima sobrepressão gerada em uma explosão de gás*. Tese (Doutorado), 2014.
- MENTER, F. R. Two-equation eddy-viscosity turbulence models for engineering applications. *AIAA Journal*, AIAA, v. 32, n. 8, p. 1598–1605, 1994.
- MOREIRA, V.; SANTOS, R.; FERREIRA, T.; VIANNA, S. Towards gilbert-johnson-keertthi algorithm distance for fluid flow numerical analysis. In: *Mechanical Engineering, 2015. Proceedings., 23 IEEE International Congress of*. [S.l.: s.n.], 2015. v. 1, p. 158–163.
- MOUKALLED F.; MANGANI, L.; DARWISH, M. *The finite volume method in computational fluid dynamics*. [S.l.]: Springer, 2016.
- PARK, D. J.; LEE, Y. S. A comparison on predictive models of gas explosions. *Korean Journal of Chemical Engineering*, Springer, v. 26, n. 2, p. 313, 2009.
- PATANKAR, S.; SPALDING, D. *A calculation procedure for the transient and steady-state behavior of shell-and-tube heat exchangers*. [S.l.]: McGraw-Hill, New York, 1974.
- PATEL, S.; JARVIS, S.; IBRAHIM, S.; HARGRAVE, G. An experimental and numerical investigation of premixed flame deflagration in a semiconfined explosion chamber. *Proceedings of the Combustion Institute*, Elsevier, v. 29, n. 2, p. 1849–1854, 2002.
- PITBLADO, R.; ALDERMAN, J.; THOMAS, J. Facilitating consistent siting hazard distance predictions using the tno multi-energy model. *Journal of Loss Prevention in the Process Industries*, Elsevier, v. 30, p. 287–295, 2014.
- POINSOT, T.; VEYNANTE, D. *Theoretical and numerical combustion*. [S.l.]: RT Edwards, Inc., 2005.

- RODI, W. Comparison of les and rans calculations of the flow around bluff bodies. *Journal of wind engineering and industrial aerodynamics*, Elsevier, v. 69, p. 55–75, 1997.
- SAETER, O. *Modelling and simulation of gas explosions in complex geometries*. [S.l.]: Norges Teknisk-naturvitenskapelige universitet, 1998.
- SARLI, V. D.; BENEDETTO, A. D.; RUSSO, G.; JARVIS, S.; LONG, E. J.; HARGRAVE, G. K. Large eddy simulation and piv measurements of unsteady premixed flames accelerated by obstacles. *Flow, turbulence and combustion*, Springer, v. 83, n. 2, p. 227–250, 2009.
- SAVILL, A.; SOLBERG, T. Some improvements to pdr/k- $\epsilon$  model predictions for explosions in confined geometries. In: *Proceedings of the IMA/ERCOFTAC Conference on Flow and Dispersion Through Groups of Obstacles. IMA Conference Series*. [S.l.: s.n.], 1994. v. 62, p. 227–249.
- SHA, W.; LAUNDER, B. E. *Model for turbulent momentum and heat transport in large rod bundles.[LMFBR]*. [S.l.], 1979.
- VERSTEEG, H. K.; MALALASEKERA, W. *An introduction to computational fluid dynamics: the finite volume method*. [S.l.]: Pearson Education, 2007.
- VIANNA, S. S.; CANT, R. S. Modified porosity approach and laminar flamelet modelling for advanced simulation of accidental explosions. *Journal of loss prevention in the process industries*, Elsevier, v. 23, n. 1, p. 3–14, 2010.
- VIANNA, S. S.; CANT, R. S. Initial phase modelling in numerical explosion applied to process safety. *Process Safety and Environmental Protection*, Elsevier, v. 92, n. 6, p. 590–597, 2014.
- WANG, M.; QIU, Y.; YUE, S. Similitude laws and modelling experiments of explosion cratering in multi-layered geotechnical media. *International Journal of Impact Engineering*, Elsevier, 2017.
- WATTERSON, J. K.; CONNELL, I. J.; SAVILL, A. M.; DAWES, W. N. A solution-adaptive mesh procedure for predicting confined explosions. *International journal for numerical methods in fluids*, Wiley Online Library, v. 26, n. 2, p. 235–247, 1998.
- WILCOX, D. C. Reassessment of the scale-determining equation for advanced turbulence models. *AIAA Journal*, AIAA, v. 96, p. 1299–1310, 1988.
- WILCOX, D. C. *et al. Turbulence modeling for CFD*. [S.l.]: DCW industries La Canada, CA, 1993. v. 2.
- YOO, J. Y.; CHOI, H. C.; HAN, S. M. Numerical analysis of turbulent flow over a backward-facing step using reynolds stress closure model. *KSME Journal*, Springer, v. 3, n. 1, p. 31–37, 1989.

# Appendix A

This Appendix A presents the logical sequence and relationship among the different subroutines making up the Porosity Flow Solver PFS as illustrated in Figure 67.

## Data Acquisition

The "Data Acquisition" comprises the first step when running PFS code. In this step the code will read the "set up" file and the PrePro output files: "mesh.parameters" and "porosity". Three different subroutines are included here: "read data", "read parameters" and "read porosity"

### Read Data

The subroutine "*Read Data*" reads data from the input set up file. The set up file is named "*flow*" and contains the flow conditions and the simulation parameters as shown in table 3.

### Read Parameters

The "*Read Parameters*" subroutine reads the mesh parameters from the PrePro output file identified by the case name with the extension "*.parameters.dat*". This is a text file containing the mesh parameters are as follows:

- **x, y, z**: overall index size of the computational domain.
- **gapx, gapy, gapz**: size of the computational cells.
- **xtrans, ytrans, ztrans**: computational domain offset based on the origin of the coordinate system.
- **DNI, DNJ, DNK**: number of computational cells in each direction.
- **NI, NJ, NK**: number of grid points in each direction.
- **numcell**: total number of computational cells.

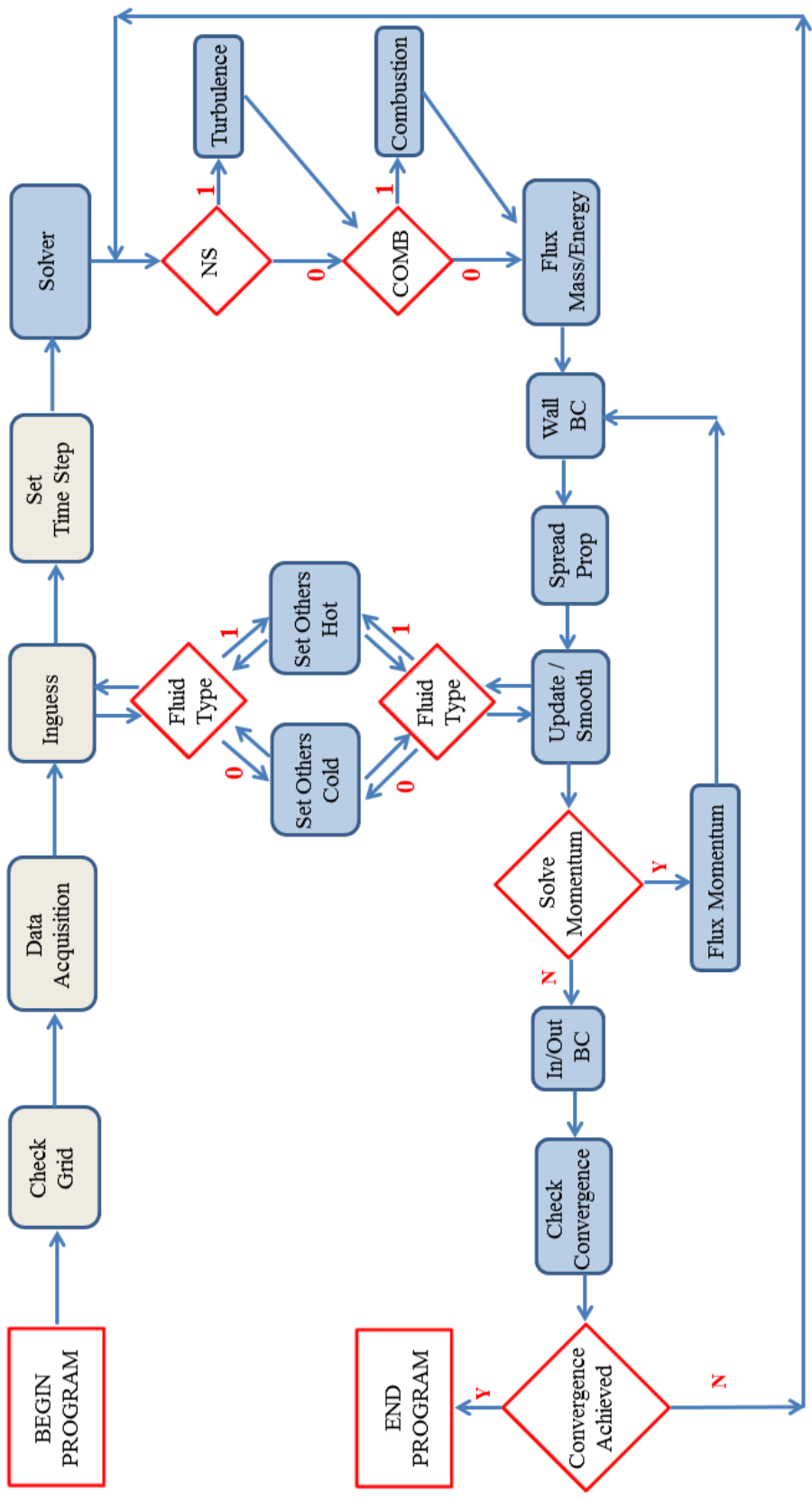


Figure 67 – Logical sequence and relationship between the different subroutines of the developed PFS code.

## Read Porosity

The subroutine "*Read Porosity*" reads the porosity files from PrePro. These are also text files and they contain the porosity values attached to each computational cell when running PrePro. The files are identified by the case name with the extensions ".porosity.asc", ".porosity.areaX.asc", ".porosity.areaY.asc" and ".porosity.areaZ.asc". The contents of these files are the volume porosity, the area porosity in "x" direction, the area porosity in "y" direction and the area porosity in "z" direction, respectively.

## Check Grid

In the subroutine "Check Grid" the computational grid is rebuild based on the mesh parameters given by the "parameters" file. The computational cells are defined as well as it is identified the connectivities between nodes, edges, faces and neighbor cells. The cells face areas and volumes are also computed here.

Once the computational cells are defined, the porosity values are attached to each one based on the content of the porosity files.

## Inguess

The subroutine "*Inguess*" provides initial values for the main flow variables in all grid. These variables are defined based on the fluid properties and flow features given by the set up "flow" input file.

The main variable defined here are as follows:

- **P**: initial pressure of the system, taken as the reference pressure  $P_{ref}$ .
- **RO**: initial fluid density, obtained by using isentropic correlations.
- **E**: initial fluid internal energy.
- **Vx, Vy, Vz**: velocity components in each direction. By default, the values of Vy and Vz are taken to zero, while Vx is calculated based on the pressure field (difference between  $P_{ref}$  and  $P_{down}$ )
- **TK, TE**: values of turbulent kinetic energy (TK) and eddy dissipation (TE) are set in accordance with suggested by (ARNTZEN, 1998).
- **FF, PV**: values of fuel fraction (FF) and reaction progress variable (PV). The value of FF is set according to the fuel type, based on its molecular weight, while PV is set to be zero.

It is important to underline here that, as the Favre average is used in the developed code, the variables  $E$ ,  $V_x$ ,  $V_y$ ,  $V_z$ ,  $TK$ ,  $TE$ ,  $FF$  and  $PV$  are weighted by the fluid density, so their values are multiplied by "RO".

Once the main variables are defined, the secondary variables are obtained by using the subroutine "*Set Others*".

## **Set Others**

The "*Set Others*" routine calculates the secondary flow variables (temperature, static pressure and the stagnation enthalpy) from the primary ones at every grid point.

This routine assumes two different types: "*Set Others Cold*" and "*Set Others Hot*". The first one is used for cold flow simulation (without combustion), while the second one is used for hot flow simulation (with combustion). The choice of which subroutine will be applied depending on the "Fluid Type" specified at the set up file.

For cold flow simulation, the "Fluid Type" in the set up file must be zero. Thus the subroutine "*Set Others Cold*" takes the air properties and solves the traditional equations for compressible flows.

For combustion simulation, the "Fluid Type" in the set up file must be an integer non-zero number: one (1) or two (2). The number (1) must be set when using methane in the combustion simulation, while the number (2) indicates that propane will be used. In this configuration, the "*Set Others Hot*" subroutine is used and solves the equations given in section 3.5.1.1 to obtain the secondary variables.

## **Set Time Step**

This routine calculates the Time Step  $\Delta t$  for the simulation based on the minimum mesh size and the CFL number (as shown in equation 4.15).

## **Solver**

In the "*Solver*" routine the iteration process is performed. It comprises the main loop and the sequence of subroutines that are assembled in order to solve the discretized differential equations. The "*Solver*" routine returns the final flow solution to the main program.

## **Turbulence**

The subroutine "*Turbulence*" is performed when setting "1" to the "NS" term in the set up file. It calculates the viscous stress and adds it to the diffusion terms of the



conservation equations. The turbulence kinetic energy and eddy dissipation source terms are also computed here.

## Combustion

The subroutine "*Turbulence*" is performed when setting "1" to the "COMB" term in the set up file. Therefore, the combustion model equations (given in section 3.5.1.2) are computed. The reaction rate progress variable and fuel fraction source terms are also obtained here.

## Flux Mass/Energy

The convective fluxes of mass and energy are computed here, into the subroutine "*flux - ma - energy*".

## Wall BC

The wall boundary conditions are set up by performing two subroutines: "*wall-bc*" and "*wall - bc - por*". In both cases a solid surface (with zero normal fluxes) is imposed, as explained in section 4.3. However, in the first case this condition is applied at the computational domain boundaries set as wall, while in the second case, the condition is defined at the computational cells attached with porosity values equal to zero.

It is important to underline here that two different index can be attached when performing the wall routines: "1" and "2". The index "1" is used in the first application of the subroutines, just after the computation of mass and energy convective fluxes, so the wall condition is defined for the variables **RO**, **E**, **FF** and **PV**. The index "2" is used in the second application of the subroutines, after the computation of momentum convective fluxes, so the wall condition is defined for the variables **VX**, **VY**, **VZ**, **TK** and **TE**.

## Spread Prop

The subroutine "*Spread Prop*" calculates the changes in the variable "PROP" (**RO**, **E**, **FF**, **PV**, **VX**, **VY**, **VZ**, **TK** and **TE**) based on the time step.

## Update / Smooth

The subroutine "*Update*" distributes the changes in each variable to the grid nodes.

The subroutine "*Smooth*" adds artificial viscosity to each variable based on the value of Smooth Factor (SF) given in the set up file.

## Flux Momentum

The convective fluxes of momentum, turbulent kinetic energy and eddy dissipation are computed here, into the subroutine *"flux - mon"*. This routine is preceded by the subroutine *"Resistance"* which calculates the resistance that unresolved obstacles offer to a mean flow.

## In / Out BC

The inflow and outflow boundary conditions are specified in the subroutine *"apply - bc"*. The values of the variables are defined according to the section 4.3. This routine also presents an additional function *"normal - v"* that removes the normal components of velocity at the wall boundaries.

## Check Convergence

The subroutine *"check - conv"* checks the changes in all primary variables over the last 5 (five) steps. The convergence criteria defined by the user at the set up input file is compared with the variables changes and a residual is calculated as shown in equation 4.16.

## Output

The *"output"* routine writes the flow solution in a *".vtk"* for use by the Kitware Paraview Program. The frequency of writing flow solution is defined by the user at the "DUMP" term of the input set up file.

Figure 68 presents a summary of the subroutines in the developed code and the main link between them.

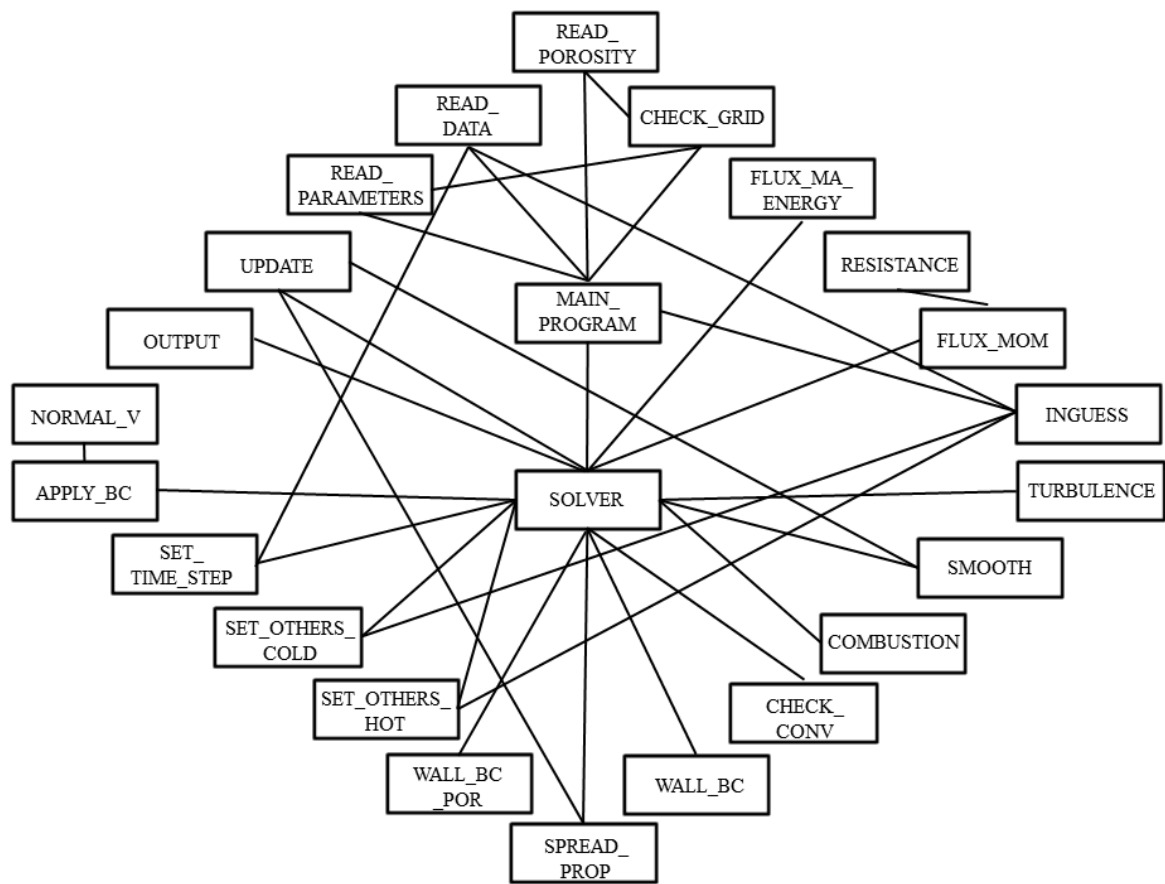


Figure 68 – Routines of the developed code and the main link between them.

# Appendix B

Appendix B comprises examples of the application of the technique developed in the current research. There have been presented results for the application of the GJK algorithm and how it works through utilisation of the stereolithography CAD file (STL) and the hexahedral mesh. The appendix also presents the flow solution calculated using the GJK methodology coupled to the modified finite volume method (FVM).

Figures 69 to 71 present three examples of the application of the developed method to calculate the flow over complex geometries.

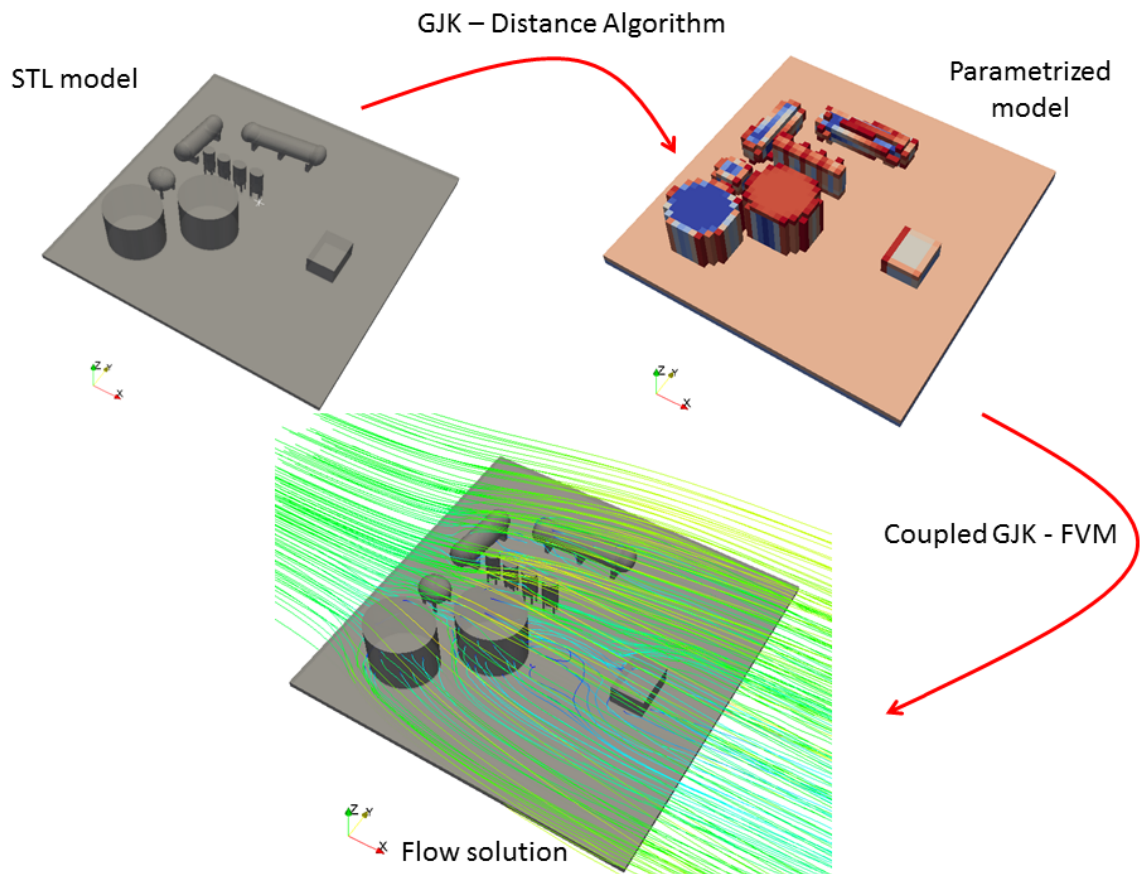


Figure 69 – Parametrization of a industrial module as porous model by using the GJK algorithm and its corresponding flow solution for an onshore site

Figures 72 and 73 present an example of the application of the developed method to explosion simulation in a complex geometry.

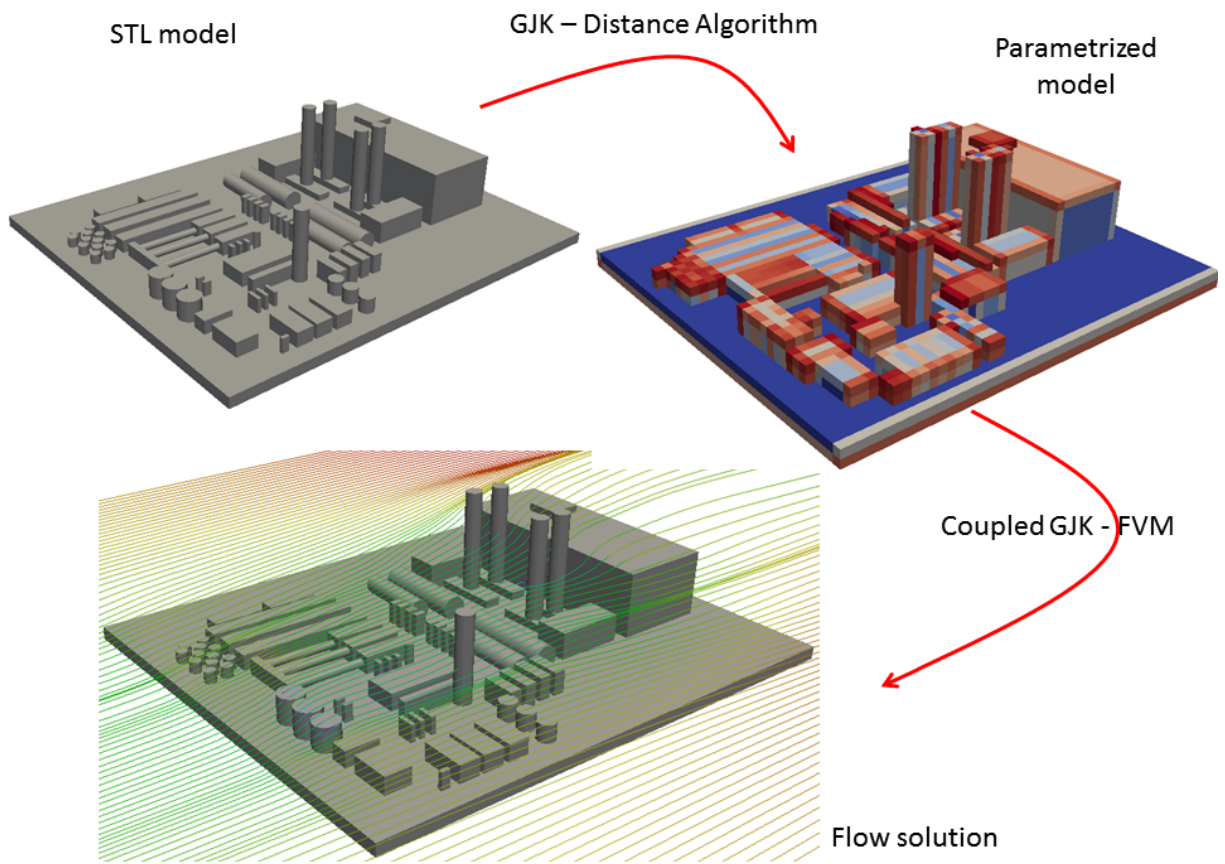


Figure 70 – Parametrization of process deck and associated flow solution.

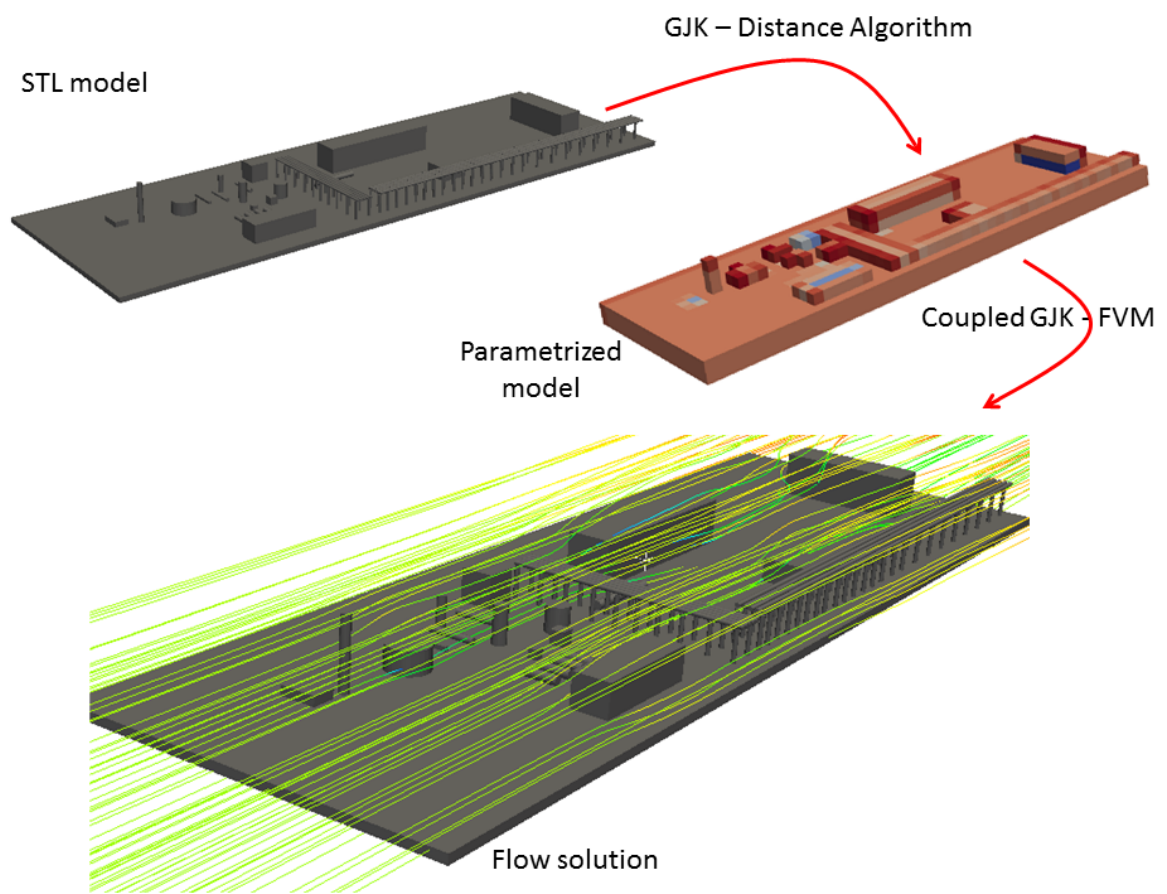


Figure 71 – Parametrization of a gas plant including details of the pipe-rack. Lower plot shows the streamlines for the velocity field.

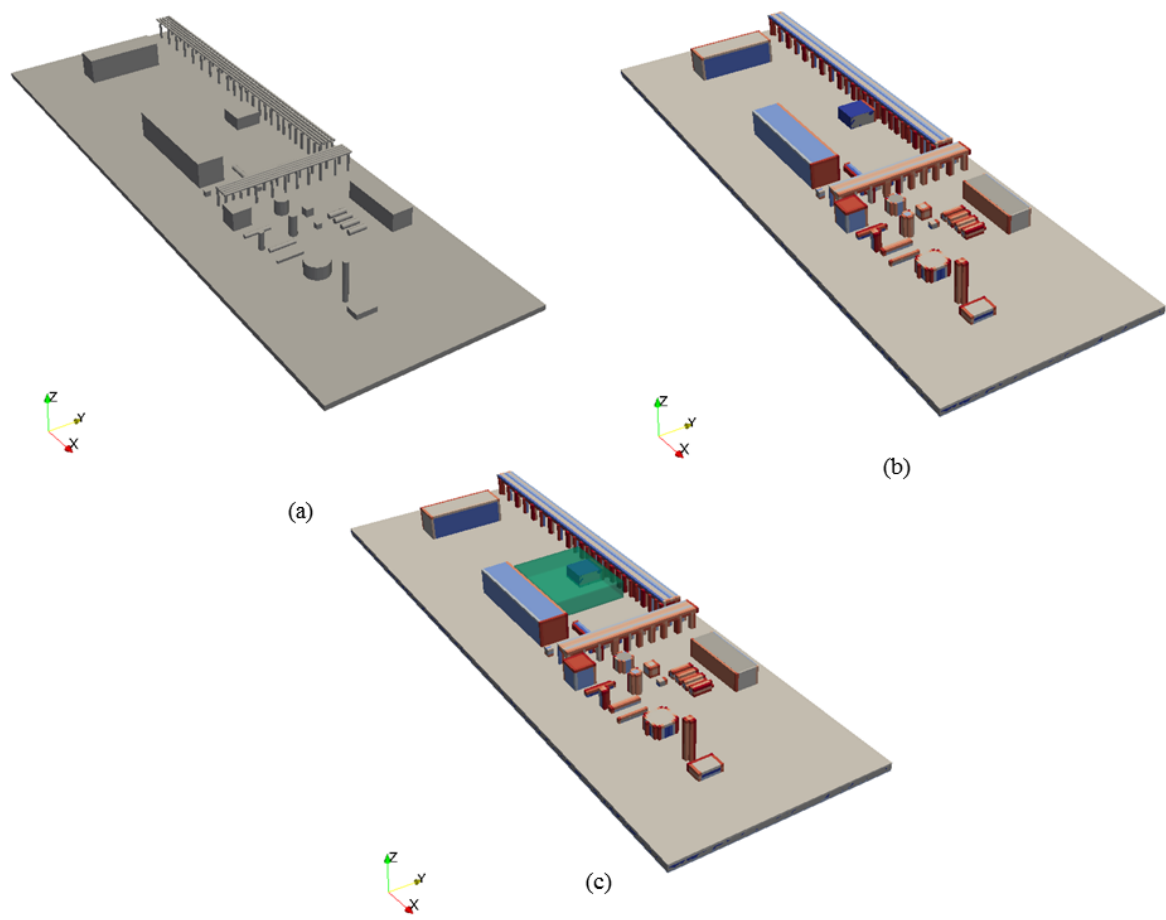


Figure 72 – Parametrization of a gas plant including details of the pipe-rack. Lower plot shows a flammable cloud (green region) inside the process area.

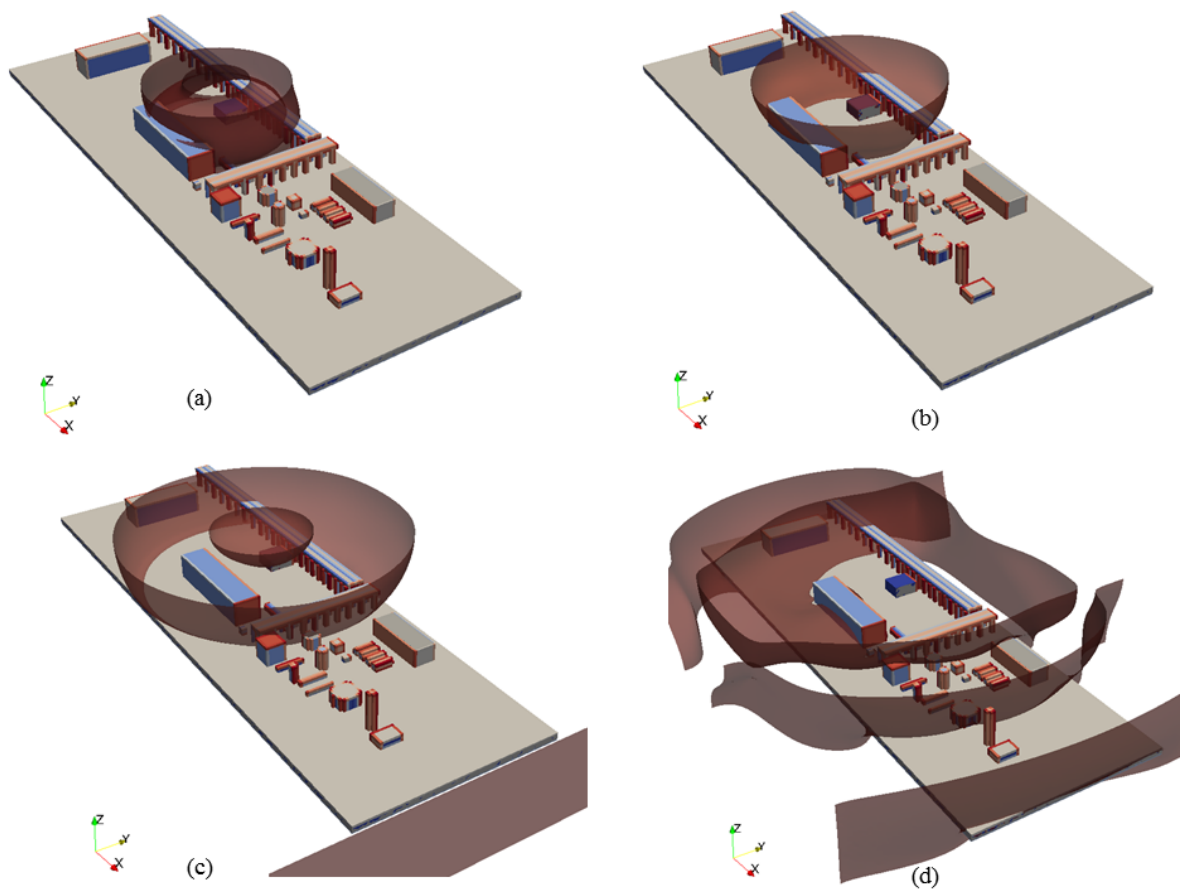


Figure 73 – Explosion simulation in a process area. The pressure wave is shown at different times after the ignition: (a)  $t = 35\text{ms}$ ; (b)  $t = 70\text{ms}$ ; (c)  $t = 105\text{ms}$ ; (d)  $t = 180\text{ms}$ .