

**UNIVERSIDADE ESTADUAL PAULISTA**  
**"JÚLIO DE MESQUITA FILHO"**  
**CAMPUS DE GUARATINGUETÁ**

**MATEUS DIAS RIBEIRO**

**Fuel Spray Modeling for Application in Internal Combustion Engines**

Guaratinguetá

2019

**Mateus Dias Ribeiro**

**Fuel Spray Modeling for Application in Internal Combustion Engines**

Tese apresentada à Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, para a obtenção do título de Doutor em Engenharia Mecânica na área de Energia.

Orientador: Prof<sup>o</sup> Dr. José Antônio Perrella Balestieri

Coorientador: Prof<sup>o</sup> Dr. Maurício Araújo Zanardi

Coorientador: Prof<sup>o</sup> Dr. Alex Mendonça Bimbato

Guaratinguetá

2019

Ribeiro, Mateus Dias  
R354f Fuel spray modeling for application in internal combustion engines / Mateus  
Dias Ribeiro - Guaratinguetá, 2019.  
149 f. : il.  
Bibliografia: f. 103-112

Tese (Doutorado) – Universidade Estadual Paulista, Faculdade de  
Engenharia de Guaratinguetá, 2019.

Orientador: Prof. Dr. José Antônio Perrella Balestieri

Coorientadores: Prof. Dr. Maurício Araújo Zanardi; Prof. Dr. Alex  
Mendonça Bimbato

1. Motores de combustão interna. 2. Fluidodinâmica computacional. 3.  
Inteligência artificial. I.Título.

CDU 621.43(043)


Pâmella Benevides Gonçalves  
Bibliotecária/CRB-8/9203

**MATEUS DIAS RIBEIRO**

ESTA TESE FOI JULGADA ADEQUADA PARA A OBTENÇÃO DO TÍTULO DE  
“DOUTOR EM ENGENHARIA MECÂNICA”


PROGRAMA: ENGENHARIA MECÂNICA  
ÁREA : ENERGIA

APROVADA EM SUA FORMA FINAL PELO PROGRAMA DE PÓS-GRADUAÇÃO

  
Prof. Dr.ª Ivonete Ávila  
Coordenadora

**BANCA EXAMINADORA:**

  
Prof. Dr. JOSÉ ANTONIO PERRELLA BALESTIERI  
Orientador / UNESP/FEG

  
Prof. Dr. JOSÉ ALEXANDRE MATELLI  
UNESP/FEG

  
Prof. Dr. JOÃO ANDRADE DE CARVALHO JUNIOR  
UNESP/FEG

*Participou por videoconferência*  
Prof. Dr. MARCIO TEIXEIRA MENDONÇA  
CTA/IAE

*Participou por videoconferência*  
Prof. Dr. WALDIR LUIZ RIBEIRO GALLO  
UNICAMP

## **DADOS CURRICULARES**

### **MATEUS DIAS RIBEIRO**

**NASCIMENTO** 08/11/1989 - Frutal / MG

**FILIAÇÃO** Ubiratã Leal Ribeiro  
Liliane de Melo Ribeiro

**2008 / 2013** Curso de Graduação em Engenharia Mecânica na Universidade Estadual Paulista, Campus de Guaratinguetá.

**2013 / 2015** Curso de Mestrado em Engenharia Mecânica na Universidade Estadual Paulista, Campus de Guaratinguetá.

**2015 / 2019** Curso de Doutorado em Engenharia Mecânica na Universidade Estadual Paulista, Campus de Guaratinguetá.

To my wife, Nicole, for always supporting me along the way.

## ACKNOWLEDGEMENTS

A PhD work is no easy endeavor, and I would not have made it to the end without the help of important people, who I wish to acknowledge here.

First of all, I am grateful to my advisors, Prof. José Antônio Perrella Balestieri, Prof. Maurício Araújo Zanardi, and Prof. Alex Mendonça Bimbato, for their guidance, teaching, and for providing the necessary tools for me to start this research. In particular, Prof. Perrella's keen sense of humor and his support during the beginning of my academic career made me a stronger person and gave me the required confidence to tackle the challenges posed in this work.

Additionally, Prof. David Schmidt deserves all of my thanks for hosting me as a visiting scholar in his lab at the University of Massachusetts Amherst. Without the resources he has provided and, more importantly, the on point balance of freedom and guidance from someone who has made so many contributions to the field of multiphase flows in CFD, I would never have accomplished what I did.

I also would like to thank my family, my mother Liliane, my father Ubiratã, and my brother Marcel. My beloved wife Nicole deserves special thanks for supporting me all the way from the beginning to the end, for listening to all my complaints patiently, for giving me strength to carry on every time I doubted I could make it to the end, and for providing free reviews of my thesis, even though she knew very little about its subject. Well, we made it to the end of it, and that wouldn't have happened without you. I love you even more now that you can explain complicated subjects, such as the difference between LES and RANS turbulence modeling approaches.

Last but not least, I would like to give a huge thanks to all the friends I made along the way, be it in Brazil or in the US. In particular, my Brazilian friends from graduate school Felliipe and Marília, for the interesting discussions about engineering, science, and politics, and of course, for restarting my Brazilian workstation every time it shut down due to a power outage during my time in the US. I am also grateful to all my colleagues at UMass Amherst, including Sampath, Hannah, Brandon, Alden, Piotr, Volodymyr, Peetak, Chinmoy, Gabe, Gavin, and Felipe. You helped me a lot with fun discussions and brilliant ideas, as well as useful technical support, which went way further than the usual "have you tried turning it off and on again?"

Este trabalho contou com o apoio das seguintes entidades:

Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES)

Código de Financiamento 001

FAPESP - Fundação de Amparo à Pesquisa do Estado de São Paulo

Processos: 2015/10299-9 (BP-DR) e 2017/04619-6 (BEPE)



*[...] the computer disease that anybody who works with computers now knows about. It's a very serious disease and it interferes completely with the work. The trouble with computers is you **play** with them. They are so wonderful.*  
*(Richard P. Feynman)*

## RESUMO

Motores de ignição a centelha com injeção direta (*direct injection spark ignition engines, DISI engines*) visam reduzir o consumo específico de combustível e respeitar os restritos níveis de emissão em motores de combustão interna de última geração. Assim, pretende-se com este trabalho desenvolver código para simulação do escoamento interno em motores DISI, assim como os fenômenos de injeção de combustível no interior da câmara de combustão utilizando uma abordagem Lagrangeana-Euleriana para representação do escoamento multifásico e Simulação de Grandes Escalas (*Large-eddy simulation, LES*) para a modelagem da turbulência no meio contínuo, por intermédio da biblioteca CFD de código aberto OpenFOAM. De modo a validar os resultados e os modelos desenvolvidos, dados experimentais serão utilizados, obtidos do motor óptico de Darmstadt, e do caso de teste de injeção de gasolina não-reativo “*Spray G*”, juntamente com o caso reativo “*Spray A*” da Rede de Combustão em Motores (*Engine Combustion Network, ECN*). Enfim, um novo código aberto será proposto para simular o motor óptico de Darmstadt em condições de escoamento a frio (sem combustão) e com combustão em condição de mistura estratificada, usando dados compilados pelo Workshop do Motor de Darmstadt (*Darmstadt Engine Workshop, DEW*) para validação. Além disso, uma abordagem de aprendizado profundo (*deep learning*) será apresentada para treinar uma rede neural artificial (*artificial neural network, ANN*) com dados de simulação LES de motores gerados neste trabalho, para realizar previsões sobre o comportamento das pequenas escalas de turbulência.

**PALAVRAS-CHAVE:** Spray. CFD. LES. Motor de Combustão Interna. Abordagem Lagrangeana-Euleriana. OpenFOAM. ANN.

## ABSTRACT

Direct injection spark ignition (DISI) engines aim at reducing specific fuel consumption and achieving the strict emission standards in state of the art internal combustion engines. Therefore, in this work the goal is to develop code for simulations of the internal flow in DISI engines, as well as the phenomenon of fuel spray injection into the combustion chamber using a Lagrangian-Eulerian approach for representing the multiphase flow, and Large-eddy Simulations (LES) for modeling the turbulence of the continuum medium by means of the open-source CFD library OpenFOAM. In order to validate the obtained results and the developed models, experimental data from the Darmstadt optical engine, and the non-reactive “Spray G” gasoline injection case, along with the reactive “Spray A” case from the Engine Combustion Network (ECN) will be employed. Finally, a novel open-source solver will be proposed to simulate the Darmstadt optical engine in motored and fired operation under stratified mixture condition, using data compiled by the Darmstadt Engine Workshop (DEW) for validation. Moreover, a deep learning framework is presented to train an artificial neural network (ANN) with the engine LES data generated in this work, in order to make predictions of the small scale turbulence behavior.

**KEYWORDS:** Spray. CFD. LES. Internal Combustion Engine. Lagrangian-Eulerian approach. OpenFOAM. ANN.

## LIST OF FIGURES

Figure 1	Comparisons of the potential of $CO_2$ reduction: ICE vs hybrid vs electric . . . .	24
Figure 2	Illustration of a V6 internal combustion engine . . . . .	27
Figure 3	Schematic representation of the four stroke cycle . . . . .	28
Figure 4	Schematic diagram of a typical fuel injector . . . . .	29
Figure 5	Example of an artificial neural network with two hidden layers . . . . .	38
Figure 6	Example of a CNN making flow predictions after learning from CFD simulation data . . . . .	38
Figure 7	An ICE simulation showing the multi-scale nature of the problem . . . . .	39
Figure 8	Velocity boundary layer over a plate and transition from laminar to turbulent . .	41
Figure 9	Energy spectrum of turbulent cascade . . . . .	42
Figure 10	Schematic representation of a full-cone spray and the different phenomena undergone by a fuel-spray during injection . . . . .	44
Figure 11	Droplet breakup regimes according to Wierzba (WIERZBA, 1990) . . . . .	47
Figure 12	Schematic representation of OFICenet . . . . .	52
Figure 13	Fully connected layers . . . . .	53
Frame 1	OFICenet algorithm . . . . .	55
Figure 14	The Darmstadt engine cylinder head variants. (a) Wall-guided and (b) spray-guided	56
Figure 15	Engine Combustion Network optically accessible combustion vessel . . . . .	57
Figure 16	Spray G injector . . . . .	58
Figure 17	Spray A injector . . . . .	59
Figure 18	Sampling planes . . . . .	60
Figure 19	Velocity field along midplane at 270 CAD bTDC . . . . .	61
Figure 20	Velocity field along midplane at 180 CAD bTDC . . . . .	62
Figure 21	Velocity field along midplane at 90 CAD bTDC . . . . .	63
Figure 22	Velocity field along midplane at 45 CAD bTDC . . . . .	64
Figure 23	Velocity, temperature, and pressure fields along valves at 270 CAD bTDC . . .	64
Figure 24	Velocity, temperature, and pressure fields along valves at 180 CAD bTDC . . .	65
Figure 25	Velocity, temperature, and pressure fields along valves at 180 CAD aTDC . . .	65
Figure 26	Velocity, temperature, and pressure fields along valves at 270 CAD aTDC . . .	66
Figure 27	Sampling lines and PDF region . . . . .	67
Figure 28	Vertical profiles at 270 CAD bTDC . . . . .	68
Figure 29	Vertical profiles at 180 CAD bTDC . . . . .	68
Figure 30	Horizontal profiles at 270 CAD bTDC . . . . .	69
Figure 31	Horizontal profiles at 180 CAD bTDC . . . . .	69
Figure 32	Vertical profiles at 90 CAD bTDC . . . . .	70
Figure 33	Vertical profiles at 45 CAD bTDC . . . . .	71
Figure 34	Horizontal profiles at 90 CAD bTDC . . . . .	71

Figure 35	Horizontal profiles at 45 CAD bTDC . . . . .	72
Figure 36	PDF of velocity components during compression stroke . . . . .	72
Figure 37	Pressure trace in averaged LES vs. averaged experiment . . . . .	73
Figure 38	Pressure trace cyclic variations . . . . .	73
Figure 39	Tumble vortex center tracking . . . . .	75
Figure 40	Tumble core from 180 to 130 CAD bTDC . . . . .	75
Figure 41	Tumble core during and after intake valves closure . . . . .	76
Figure 42	Tumble core reduction during compression stroke . . . . .	77
Figure 43	Tumble core break-down . . . . .	77
Figure 44	Two point velocity correlation . . . . .	78
Figure 45	Length scale of coherent turbulent structures . . . . .	79
Figure 46	M field . . . . .	80
Figure 47	M plot . . . . .	80
Figure 48	Spray simulation at 0.3 ms . . . . .	81
Figure 49	Spray simulation at 0.6 ms . . . . .	81
Figure 50	Spray simulation at 1.0 ms . . . . .	82
Figure 51	Spray simulation at 1.4 ms . . . . .	82
Figure 52	Penetration length for liquid/gas phases validated against experimental data . . . . .	83
Figure 53	Number of parcels/evaporated mass and droplet size along time . . . . .	84
Figure 54	Sampling plane at 22.5°, as defined by Sphicas et al. (2017b) . . . . .	84
Figure 55	Gas axial velocity at 22.5° plane . . . . .	85
Figure 56	Definition of sampling lines for comparison between simulation and experiment . . . . .	85
Figure 57	(a) SMD in radial direction at 0.5 ms ASOI, (b) SMD in transverse direction at 0.5 ms ASOI . . . . .	86
Figure 58	(a) SMD in radial direction at 0.6 ms ASOI, (b) SMD in transverse direction at 0.6 ms ASOI . . . . .	86
Figure 59	(a) Axial droplet velocity in radial direction at 0.5 ms ASOI, (b) Axial droplet velocity in transverse direction at 0.5 ms ASOI . . . . .	87
Figure 60	(a) Axial droplet velocity in radial direction at 0.6 ms ASOI, (b) Axial droplet velocity in transverse direction at 0.6 ms ASOI . . . . .	87
Figure 61	Temperature and iso-octane/carbon monoxide mass fractions at the central plane for the PRF100 case as ignition starts to unfold at 6 ms . . . . .	88
Figure 62	Penetration length for both liquid and gas phases . . . . .	88
Figure 63	Ignition delay against experimental data . . . . .	89
Figure 64	Flame lift-off length validated against experimental data . . . . .	89
Figure 65	Velocity magnitude during fuel injection from 75 CAD bTDC to 60 CAD bTDC . . . . .	91
Figure 66	Spray parcels and velocity, fuel mass fraction, and temperature fields . . . . .	91
Figure 67	Iso-octane mass fraction field and velocity vector plot at spark timing . . . . .	92
Figure 68	Temperature at flame front from 15 CAD bTDC to 20 CAD aTDC . . . . .	92
Figure 69	Flame quenching . . . . .	93

Figure 70	Comparison of PDFs of the velocity field with and without late injection . . . .	94
Figure 71	Pressure trace of fired GDI simulation . . . . .	94
Figure 72	Temperature trace of fired GDI simulation . . . . .	95
Figure 73	Gas density trace of fired GDI simulation . . . . .	95
Figure 74	Species mass fraction concentrations along time . . . . .	96
Figure 75	Calculation of turbulent kinematic viscosity using CFD and deep learning . . .	97
Figure 76	Prediction from CFD (left) against prediction from deep neural network (right)	98
Figure 77	Control volume . . . . .	114
Figure 78	Shear forces on volume element . . . . .	116
Figure 79	One-dimensional control volume . . . . .	118
Figure 80	Conventional Grid . . . . .	122
Figure 81	Staggered Grid vs Collocated Grid . . . . .	123
Figure 82	Engine simulation folders . . . . .	126
Figure 83	Mesh generation process at 45 bTDC - initial domain generation . . . . .	127
Figure 84	Mesh generation process at 45 bTDC - final mesh . . . . .	127
Figure 85	Spray G simulation computational mesh . . . . .	129
Figure 86	Velocity PDFs at 270, 180, 90, and 45 CAD bTDC . . . . .	131
Figure 87	PDF of velocity gradients . . . . .	131
Figure 88	PDF of turbulent kinetic energy . . . . .	132
Figure 89	PDF of 2D shear . . . . .	132
Figure 90	PDF of 2D vorticity . . . . .	132
Frame 2	OFICenet Training Log . . . . .	133
Frame 3	GDIidict dictionary . . . . .	136
Frame 4	engineGeometry dictionary . . . . .	138
Frame 5	dynamicMeshDict dictionary . . . . .	139
Frame 6	sprayCloudProperties dictionary . . . . .	139
Frame 7	chemistryProperties dictionary . . . . .	143
Frame 8	combustionProperties dictionary . . . . .	143
Frame 9	CHEMKIN mechanism file for the single-step reaction of iso-octane/n-heptane	144
Frame 10	NASA polynomials of species involved in the single-step reaction mechanism of iso-octane/n-heptane . . . . .	145
Frame 11	Reduced reaction mechanism of Zhou et al. (2016) . . . . .	145

## LIST OF TABLES

Table 1 – Transition Weber numbers for different regimes . . . . .	48
Table 2 – Reaction rate parameters for a few selected fuels. Units are cm-s-mole-kcal-Kelvin	50
Table 3 – Some selected forward reactions of Zhou et al’s mechanism . . . . .	51
Table 4 – Information on OFICEnet architecture and solution strategy . . . . .	54
Table 5 – Darmstadt engine, (BAUM et al., 2014) . . . . .	57
Table 6 – “Spray G” condition . . . . .	58
Table 7 – Modified “Spray A” condition . . . . .	59
Table 8 – Discretized equation coefficients . . . . .	120
Table 9 – Pressure correction equation coefficients . . . . .	124
Table 10 – Valve timing . . . . .	128
Table 11 – Summary of boundary conditions . . . . .	128
Table 12 – Spray-guided Darmstadt engine hollow-cone injection parameters . . . . .	130
Table 13 – Information on the GDI engine fired case . . . . .	130

## LIST OF ABBREVIATIONS AND ACRONYMS

ANL	Argonne National Laboratories
ANN	Artificial Neural Network
ASOI	After Start Of Injection
AVBP	A Very Big Project
aTDC	After TDC (unit of engine time)
BDC	Bottom Dead Center
bTDC	Before TDC (unit of engine time)
CAD	Crank Angle Degree
CDF	Cumulative Density Function
CDS	Central Differencing Scheme
CFD	Computational Fluid Dynamics
CFL	Courant-Friedrich-Lewy number
CHEMKIN	Chemical Kinetics
CNN	Convolutional Neural Network
DEW	Darmstadt Engine Workshop
DISI	Direct injection spark ignition
DNN	Deep Neural Network
DNS	Direct Numerical Simulation
DOE	United States Department of Energy
ECN	Engine Combustion Network
EGR	Exhaust Gas Recirculation
EPA	United States Environmental Protection Agency
FVM	Finite Volume Method
GAMG	Generalised Geometric-Algebraic Multi-Grid solver
GDI	Gasoline Direct Injection



GPL	GNU General Public License
ICCT	International Council on Clean Transportation
ICE	Internal Combustion Engine
LES	Large-eddy simulation
LIF	Laser Induced Fluorescence
OFICE	OpenFOAM solver for Internal Combustion Engine simulations
OFICenet	Internal Combustion Engine neural Network
OPEC	Organization of Arab Petroleum Countries
OpenFOAM	Open Field Operation and Manipulation
PaSR	Partially Stirred Reactor
PBiCG	Preconditioned bi-conjugate gradient (PBiCG)
PDF	Probability Density Function
PFI	Port-Fuel Injection
PISO	Pressure Implicit with Splitting of Operator
PIV	Particle Image Velocimetry
PLIF	Planar Laser Induced Fluorescence
PRF	Primary Reference Fuel
RANS	Reynolds-Averaged Navier-Stokes
ReLU	Rectified Linear Unit
SIMPLE	Semi-Implicit Method for Pressure Linked Equations
SG	Spray-Guided
SST	Shear Stress Tensor
STL	Stereolithography
TDC	Top Dead Center
TKE	Turbulent Kinetic Energy
TVD	Total Diminishing Variation
WG	Wall-Guided

## LIST OF SYMBOLS

$A$	Pre-exponential factor
$A_i$	Reactive species
$CO_2$	Carbon Dioxide
$C_s$	Smagorinsky constant
$C_D$	Drag coefficient
$c_{p,l}$	Specific heat at constant liquid fuel pressure
$c_{p,vap}$	Specific heat at vaporizing pressure
$C_{bag}$	Reitz-Diwakar breakup model bag constant
$C_{strip}$	Reitz-Diwakar breakup model stripping constant
$C_1$	Reitz-Diwakar breakup constant
$C_2$	Reitz-Diwakar breakup constant
$cm^3$	Cubic centimeter (unit volume)
$d$	Droplet diameter
$dm_{evap}$	Evaporated mass
$D_i$	Binary diffusivity coefficient
$E_A$	Activation energy
$g_i$	Gravity acceleration in direction i
$g$	Gas phase
$G_l$	Rate of vaporizing mass
$G_2$	Rate of vaporizing mass for flash boiling condition
$G_f$	Flash boiled vapor rate
$h$	Enthalpy
$h$	Heat transfer coefficient
$h_{inf}$	Gaseous enthalpy in the mixture
$h_b$	Gaseous enthalpy at drop surface

$h_t$	Total enthalpy term
$k$	Turbulent Kinetic Energy
$K$	Kinetic energy
$k_f$	Forward reaction rate coefficient
$k_r$	Reverse reaction coefficient
$kg/m^3$	Kilogram per cubic meter (unit of density)
$K_r$	Resolved turbulent kinetic energy
$k_r$	Unresolved turbulent kinetic energy
$L(T_b)$	Latent heat at local boiling temperature
$L2$	L2 norm
$L_\delta$	Length scale of turbulent coherent structures
$L_C$	Connecting rod length
$m_P$	Parcel mass
$mg$	Miligram (unit of mass)
$M$	Point lying in vortex plane
$M(x, t)$	Pope LES quality criterion
$\dot{m}_{evap}$	Evaporated mass flow rate
$NO_x$	Nitrogen Oxides
$NO$	Nitric Oxide
$Nu$	Nusselt number
$N$	number of points inside the sampled vortex area
$OH$	Hydroxyl
$p$	Pressure
$P$	Parcel or liquid phase
$Pr$	Prandtl number
$P$	Vortex center
$PM$	Distance between vortex center and point lying in vortex plane

$p$	Piston position
$q$	Distance where the autocorrelation function firstly intersects with the zero line
$q_{cond}$	Thermal conductivity
$Re$	Reynolds number
$r_0$	Droplet radius
$r_{stable}$	Stable radius after droplet breakup
$R$	Universal gas constant
$R_\alpha$	Cross-correlation function
$S$	Source term
$S_{ij}$	Rate-of-strain tensor
$Sh$	Sherwood number
$S$	Vortex plane
$t$	Time
$T$	Absolute temperature
$T_{inf}$	Temperature in the gas phase
$u_i$	Velocity component in direction i
$u_{rel}$	Relative velocity droplet-gas
$u_\alpha^\delta$	Deviation of resolved velocity to the spatial mean
$U_M$	Velocity vector
$We$	Weber number
$x_i$	Spacial component i
$x$	Value within Rosin-Rammler distribution
$x_P$	Parcel position
$Y_i$	Mass fraction of species i
$Y_s$	Instantaneous mass fraction at droplet surface
$z$	Unit vector

### **Greek symbols:**

$\alpha_{eff}$	Effective diffusivity
$\beta_i$	Adam optimizer coefficients
$\gamma$	Diffusion coefficient
$\Gamma$	Vortex center parameter
$\Gamma_\phi$	General diffusion coefficient
$\delta$	Partial derivative
$\Delta$	Filter width
$\Delta h_{evap}$	Enthalpy of vaporization
$\epsilon$	Dissipation of k
$\epsilon_t$	Total energy term
$\zeta$	Dimensionless correction factor of heat/mass transfer
$\eta$	Adam optimizer step size
$\lambda_g$	Thermal conductivity of the gas mixtures
$\mu_{eff}$	Effective viscosity
$\mu m$	Micro meter (unit of space)
$\nu$	Dynamic Viscosity
$\nu_i$	Stoichiometric coefficient of reaction
$\rho$	Fluid density
$\sigma$	Droplet surface tension
$\tau_{ij}^{sgs}$	Sub-grid stresses
$\tau_{br}$	Characteristic droplet breakup time
$\Phi$	Flux
$\chi^2$	Chi-square test
$\omega$	Rate of dissipation of k
$\omega_i$	Species concentration source term
$\nabla$	Nabla operator

## TABLE OF CONTENTS

<b>1</b>	<b>INTRODUCTION</b>	<b>22</b>
1.1	GOALS AND CONTRIBUTIONS OF THIS WORK	25
1.2	OVERVIEW OF THIS WORK	26
<b>2</b>	<b>A REVIEW ON INTERNAL COMBUSTION ENGINES RESEARCH</b>	<b>27</b>
2.1	BASICS	27
2.2	EARLY DEVELOPMENT	29
2.3	EXPERIMENTAL METHODS	31
2.4	COMPUTATIONAL METHODS	33
<b>2.4.1</b>	<b>Conventional Methods: Computational Fluid Dynamics</b>	<b>33</b>
<b>2.4.2</b>	<b>Novel Methods: Artificial Intelligence</b>	<b>36</b>
<b>3</b>	<b>MODELING APPROACHES</b>	<b>40</b>
3.1	GOVERNING EQUATIONS	40
3.2	TURBULENCE MODELING	40
<b>3.2.1</b>	<b>The Smagorinsky subgrid model</b>	<b>43</b>
3.3	MODELING OF GASOLINE DIRECT INJECTION ENGINES	43
<b>3.3.1</b>	<b>Moving Mesh Strategy</b>	<b>43</b>
<b>3.3.2</b>	<b>Fuel-Spray Modeling</b>	<b>44</b>
<b>3.3.3</b>	<b>Combustion modeling</b>	<b>48</b>
<b>3.3.4</b>	<b>OFICenet</b>	<b>51</b>
<b>4</b>	<b>EXPERIMENTAL DATABASES FOR VALIDATION</b>	<b>56</b>
4.1	OPTICALLY ACCESSIBLE INTERNAL COMBUSTION ENGINE	56
4.2	OPTICALLY ACCESSIBLE COMBUSTION VESSEL	57
<b>4.2.1</b>	<b>ECN “Spray G”</b>	<b>58</b>
<b>4.2.2</b>	<b>ECN “Spray A”</b>	<b>59</b>
<b>5</b>	<b>RESULTS AND DISCUSSION</b>	<b>60</b>
5.1	ENGINE FLOW STUDY	60
<b>5.1.1</b>	<b>Flow Visualization</b>	<b>60</b>
5.1.1.1	Flow along midplane	60
5.1.1.2	Flow along valves	62
<b>5.1.2</b>	<b>Velocity Field</b>	<b>66</b>
5.1.2.1	Intake Stroke Profiles	67
5.1.2.2	Compression Stroke Profiles	70
5.1.2.3	PDF of the Velocity Field Along Midplane	72
<b>5.1.3</b>	<b>Pressure Trace</b>	<b>73</b>
<b>5.1.4</b>	<b>In-cylinder Turbulence</b>	<b>74</b>
5.1.4.1	Tumble Motion	74
5.1.4.2	Two-point Correlation	76

5.1.4.3	LES Quality Criterion . . . . .	79
5.2	FUEL SPRAY STUDY . . . . .	79
<b>5.2.1</b>	<b>Flow Visualization . . . . .</b>	<b>79</b>
<b>5.2.2</b>	<b>Results along time . . . . .</b>	<b>83</b>
5.2.2.1	Penetration Length . . . . .	83
5.2.2.2	Particle Count and Droplet Size . . . . .	83
5.2.2.3	Gas Velocity . . . . .	84
<b>5.2.3</b>	<b>Results at Specific Time-steps . . . . .</b>	<b>85</b>
5.2.3.1	Droplet Size . . . . .	85
5.2.3.2	Droplet Velocity . . . . .	86
5.3	HETEROGENEOUS MIXTURE COMBUSTION STUDY . . . . .	87
<b>5.3.1</b>	<b>Flow Visualization . . . . .</b>	<b>87</b>
<b>5.3.2</b>	<b>Penetration Length . . . . .</b>	<b>88</b>
<b>5.3.3</b>	<b>Ignition Delay . . . . .</b>	<b>88</b>
<b>5.3.4</b>	<b>Flame Lift-off Length . . . . .</b>	<b>89</b>
5.4	GASOLINE DIRECT INJECTION ENGINE STUDY . . . . .	90
<b>5.4.1</b>	<b>Flow Visualization . . . . .</b>	<b>90</b>
5.4.1.1	Fuel-injection . . . . .	90
5.4.1.2	Flame propagation . . . . .	92
<b>5.4.2</b>	<b>Impact of Fuel Spray on Turbulence . . . . .</b>	<b>93</b>
<b>5.4.3</b>	<b>Averaged results along time . . . . .</b>	<b>94</b>
5.5	ICE NETWORK STUDY . . . . .	96
<b>5.5.1</b>	<b>Training . . . . .</b>	<b>96</b>
<b>5.5.2</b>	<b>Results . . . . .</b>	<b>97</b>
<b>6</b>	<b>CONCLUSIONS . . . . .</b>	<b>99</b>
6.1	RESEARCH SUMMARY . . . . .	99
6.2	FUTURE WORK . . . . .	101
	<b>REFERENCES . . . . .</b>	<b>103</b>
	<b>APPENDIX A – GOVERNING EQUATIONS . . . . .</b>	<b>114</b>
A.1	EQUATIONS DERIVATION . . . . .	114
A.2	NUMERICAL SOLUTION . . . . .	118
	<b>APPENDIX B – NUMERICAL SETUP OF SIMULATIONS . . . . .</b>	<b>126</b>
B.1	COLD-FLOW ENGINE CASES . . . . .	126
B.2	FUEL-SPRAY CASES . . . . .	128
B.3	FIRED ENGINE CASES . . . . .	130
	<b>APPENDIX C – ADDITIONAL RESULTS . . . . .</b>	<b>131</b>
	<b>ANNEX A – SIMULATION SETUP FILES . . . . .</b>	<b>136</b>
A.1	MOVING MESH . . . . .	136
A.2	FUEL-SPRAY CLOUD . . . . .	139
A.3	COMBUSTION . . . . .	143

## 1 INTRODUCTION

Since the end of the 19th century, the internal combustion engine (ICE) has been an important piece of technology for the energy sector worldwide, especially for the purpose of transportation. The remarkable success of this technology over such a long period of time was possible due to intense research that resulted not only in an increase in thermal efficiency and power output (HEYWOOD, 1988; BASSHUYSEN; SCHÄFER, 2002; PETERSON; REUSS; SICK, 2014), but also in a reduction in harmful emissions (LIBERMAN, 2008; GORYNTSEV et al., 2010).

In addition to the effort of improving efficiency and emissions mentioned above, the field also captures the attention of researchers for the several interesting phenomena occurring during the operation of an ICE. These phenomena include a rather complicated turbulent flow generated by moving boundaries (such as valves and pistons) operating at very high speeds, a complex multiphase flow initiated by the injection of fuel at high pressure in the combustion chamber (in the case of diesel and some modern gasoline engines) or in the intake ducts (in the case of conventional gasoline engines), and finally, the combustion process, which by itself is a very complicated problem with many variables to be solved. The understanding of each of these phenomena alone are of considerable interest for engineering, and therefore, useful in many modern problems addressed by engineers today.

Among these aforementioned problems, the rising levels of greenhouse gases and the increase in harmful pollution levels since the large scale adoption of ICEs in the transportation sector are of special concern. Although the exploitation of fossil fuels allowed economies worldwide to grow rapidly since the industrial revolution (LIBERMAN, 2008), being therefore crucial for a considerable increase in people's quality of life, this uncontrolled economic growth led to public health problems and damage to the environment. In order to solve these issues without compromising the growing need for energy, solutions are being proposed to develop cleaner and more efficient ways of power generation. Within the transportation sector, electrification of powertrains have been sought as feasible alternatives to ICEs, as well as hybrid engines (electrical motors combined with internal combustion engines). However, since the current status of battery technology still holds the mileage of electric cars at low operational ranges (compared to gasoline cars) with long recharging times (ILLGEN; HÖCK, 2018), internal combustion engines will still be present for many years to come. Thus, it is important to continue deploying efforts to make ICEs even more efficient during this transitional period, especially considering that emerging markets (such as Brazil's) are normally very price sensitive, and even after a considerable improvement in the battery technology, the adoption of zero-emission<sup>1</sup> cars will likely be slow in these countries.

Yet another argument in favor of the ICE technology is illustrated in Figure 1, showing a comparison among different technologies in regard to their efficiency and  $CO_2$  emission levels. Although electric vehicles emit considerably less  $CO_2$  per mile driven compared to ICE powered vehicles (considering the US power grid), the large size of the batteries used in fully electric cars is a considerable drawback in

---

<sup>1</sup> Assuming that the electricity source used in the electric car does not come from the burning of fossil fuels, which is well beyond a reality in most countries today (EIA, 2018).



their potential to prevent  $CO_2$  emissions per kWhr of battery per year. This means that the employment of a hybrid technology can be much more beneficial to greenhouse gas emissions prevention in the short and long term, since this technology benefits from continuous improvements in both ICE and battery areas in order to make impressive fuel economy gains. For instance, as shown in Hanley (2019), using data from the US Department of Energy, the 2019 Toyota Camry LE hybrid has a battery 100 times smaller than the 2019 Tesla Model S Electric (1 kWhr against 100 kWhr), and yet the hybrid car has double the range of the electric counterpart (676 miles against 335 miles). Therefore, the hybrid vehicle shown in this analysis can prevent 1035.5 kg of  $CO_2$ /kWhr/year (considering the energy mix of all 50 US states), whereas the electric vehicle with best performance in regard to this metric can prevent only 76 kg of  $CO_2$ /kWhr/year (considering the strictly regulated state of California). This situation can be even worse if the electric vehicle in question is the Tesla Model S driven in Michigan (only 4.2 kg of  $CO_2$ /kWhr/year), where the majority of the electricity produced comes from coal-fired power plants (EIA, 2019). Similar figures can also be found in other countries, such as Germany, even though Germany has committed to increasing the share of renewable sources to its gross power generation mix (AMELANG, 2018). Buchal, Karl and Sinn (2019) have shown that driving a diesel powered Mercedes C-220-d in Munich emits less  $CO_2$  per mile driven than a Tesla Model 3 (75 kWh battery), because of the still high reliance of the country on coal-fired power plants. However, markets such as Brazil's where the vast majority of the power production comes from renewable sources (hydropower), electrified powertrains yield reduced  $CO_2$  tailgate emissions compared to ICE powered vehicles. But then, the extensive adoption of biofuels (such as ethanol) by that country may allow an even higher  $CO_2$  prevention per kWhr per year figure if the hybrid approach is employed. Therefore, due to the urgent necessity of reducing greenhouse emissions worldwide, a hybrid approach would not only ensure better  $CO_2$  emission prevention now, as it would also be a more efficient transitional solution for the transportation sector, allowing more time for the battery supply chains to mature (HANLEY, 2019) and more reliable renewable sources of electricity to be created. Since hybrid vehicles also rely on the ICE technology, this is another reason to continuously study ICE related flows.

With the purpose of pushing the industry to achieve cleaner operation in modern internal combustion engines worldwide, legislation on greenhouse and pollutant emissions has become much stricter in recent years. For instance, projections for the European  $CO_2$  emissions have a target fleet average of 95 g/km for all new passenger cars by 2021, while the United States Environmental Protection Agency (EPA) has committed to an average of 93 g/km by 2025 (ICCT, 2014). In order to accomplish these ambitious goals, more innovative concepts, such as gasoline direct injection (GDI) engines, are gaining popularity among manufacturers of four-stroke spark-ignited engines (also known as the Otto engine). The difference of this new concept in comparison to the conventional port-fuel injection (PFI) approach lies in the fact that in the former the fuel is injected directly into the combustion chamber, thus allowing more control of the fuel quantity injected in order to achieve economy (PETERSON; REUSS; SICK, 2014; GORYNTSEV et al., 2010). However, to confirm that premise, these engines must be studied regarding each of the aforementioned phenomena taking place during operation. For that intention, there are essentially two ways of performing such a study. One way is to perform an experimental study, constructing a GDI engine test-bench in a laboratory with the necessary measurement apparatuses to

Figure 1 – Comparisons of the potential of  $CO_2$  reduction: ICE vs hybrid vs electricTable. Vehicle Comparisons (data from [www.fueleconomy.gov](http://www.fueleconomy.gov))

	2017			2019					
	Toyota Camry	Toyota Camry	Toyota Camry LE Hybrid	Hyundai Ioniq Electric	Chevrolet Bolt Electric	Tesla Model S Electric			
MPG or MPGe	27	34	52	136	119	102			
Range (miles)	459	493	676	124	238	335			
MSRP-\$US	\$23,070	\$24,350	\$28,150	\$30,315	\$37,495	\$94,000			
MSRP-\$CDN <sup>1</sup>	-na-	\$27,950	\$31,550	\$37,899	\$44,300	\$124,600			
Battery Size (kWhr)	0	0	1	28	60	100			
Carbon Dioxide (CO <sub>2</sub> ) Comparisons									
State	50 state	50 state	50 state	California 90210	Michigan 48103	California 90210	Michigan 48103	California 90210	Michigan 48103
Zip code									
Tailpipe& Upstream CO <sub>2</sub> (g/mile)	395	314	205	90	200	100	230	110	270
Annual CO <sub>2</sub> @ 9500 miles annually (kg)	3752.5	2983	1947.5	855	1900	950	2185	1045	2565
CO <sub>2</sub> prevented over 9500 miles annually (kg)	0	769.5 <sup>2</sup>	1035.5 <sup>3</sup>	2128 <sup>3</sup>	1083 <sup>3</sup>	2033 <sup>3</sup>	798 <sup>3</sup>	1938 <sup>3</sup>	418 <sup>3</sup>
CO <sub>2</sub> prevented / battery kWhr /year (kg CO <sub>2</sub> /kWhr/yr)	-na-	-na-	1035.5	76	38.7	33.9	13.3	19.4	4.2

<sup>1</sup> prices from manufacturer Canadian websites as of early March 2019 and are subject to change.

<sup>2</sup> relative to the 2017 Camry

<sup>3</sup> relative to the 2018 Camry

Source: Hanley (2019).

capture information about the air/fuel flow through its intake/exhaust ducts and inside the combustion chamber, as well as to gather information about the thermodynamic state (pressure, temperature), and the evolution of the flame inside the engine. In this case, it is important for the engine to be optically accessible, i.e. engine liner and piston made of transparent material like quartz, so state-of-the-art diagnostics techniques can be applied, such as planar laser induced fluorescence (PLIF), particle image velocimetry (PIV), and Mie scattering. However, this approach is very expensive, due to the high price of measurement devices, and rather limited, since the optical window very often cannot reach the entire engine geometry, especially in regions close to complex geometric features, such as the cylinder head. The second way is to perform a numerical study, using a computational fluid dynamics (CFD) code to simulate the behaviour of a GDI engine using high performance computers, ideally validating the numerical results with experimental ones. The main advantage of this approach is that even though the license for a commercial CFD code capable of performing engine simulations can be very expensive, the cost of performing such simulations is much lower than performing experiments. In order to attest the quality of the code, validation against experimental results are still necessary during the development phase, but as soon as this stage is completed the code can be used as many times as necessary for testing several designs at low additional cost. Finally, as a result of the practicality of numerical codes and increasing performance of computers with time, larger amounts of CFD results are made available as time passes, thus creating the opportunity of using novel computational techniques in the field of artificial intelligence, such as machine learning and deep neural networks, in CFD codes, allowing faster and more accurate predictions (RAISSI; PERDIKARIS; KARNIADAKIS, 2019; TOMPSON et al., 2017; RAISSI et al., 2018).

Therefore, the primary scope of this work is to perform a numerical study of the GDI engine technology using an open-source CFD code, as well as to assess the possibility of using deep neural networks to assist the CFD code in the prediction of engine flows. In the next section, the goals and contributions of this work will be presented, followed by a quick overview of the structure of this document.

## 1.1 GOALS AND CONTRIBUTIONS OF THIS WORK

The general goal of this work is the modeling of a gasoline direct injection internal combustion engine (GDI-ICE), using an open-source CFD toolbox based on the OpenFOAM code (WELLER et al., 1998). This study considers every separate aspect of operation of a GDI-ICE, such as the turbulent flow using large eddy simulation (LES), fuel-spray modeling using a Lagrangian-Eulerian approach, and the stratified mixture combustion using reduced reaction mechanisms obtained in the literature. In order to validate the simulation results, experimental measurements from two different sources are used. For the engine flow and pressure trace along the cycle, the Darmstadt optical engine (BAUM et al., 2014) is used. For the fuel-spray simulations, regarding spray penetration length, droplet size distribution, gas/droplet velocity fields, and spray combustion, results obtained from groups participating in the engine combustion network (ECN, 2018) are chosen as validation sources.

The specific goals of the current work are:

- Study of the flow field inside the Darmstadt optical engine under motored operation;
- Study of a non-reactive fuel-spray considering all its related phenomena, such as primary and secondary atomization, droplet breakup, heat transfer, and evaporation;
- Study of reactive fuel-sprays using reaction mechanisms of varying levels of complexity;
- Validation of the code against available experimental data;

from which the main contributions are the following:

- The creation of a free and open-source CFD solver, distributed under a GPLv3 license, that is capable of making full simulations of internal combustion engines and related phenomena (turbulent flow predictions within moving mesh, fuel-injection, and combustion), making the research of these phenomena accessible to more people, especially when resources are scarce. The code is named OFICE (OpenFOAM solver for Internal Combustion Engine simulations);
- The proposal of a deep neural network model (OFICenet) for predicting properties of the small turbulent scales. This is one of the first works to study the possibility of combining deep neural network techniques with a CFD code, and, to the author's knowledge, the first to apply such techniques to the prediction of flow properties in internal combustion engines.

## 1.2 OVERVIEW OF THIS WORK

A review of literature on research methods for internal combustion engines is presented in chapter 2. In chapter 3, an overview of the numerical methods used in this work is presented, introducing the governing equations and the specific numerical models for the modeling of GDI engines, such as mesh motion, fuel-injection, and combustion, as well as the above mentioned OFICenet architecture.

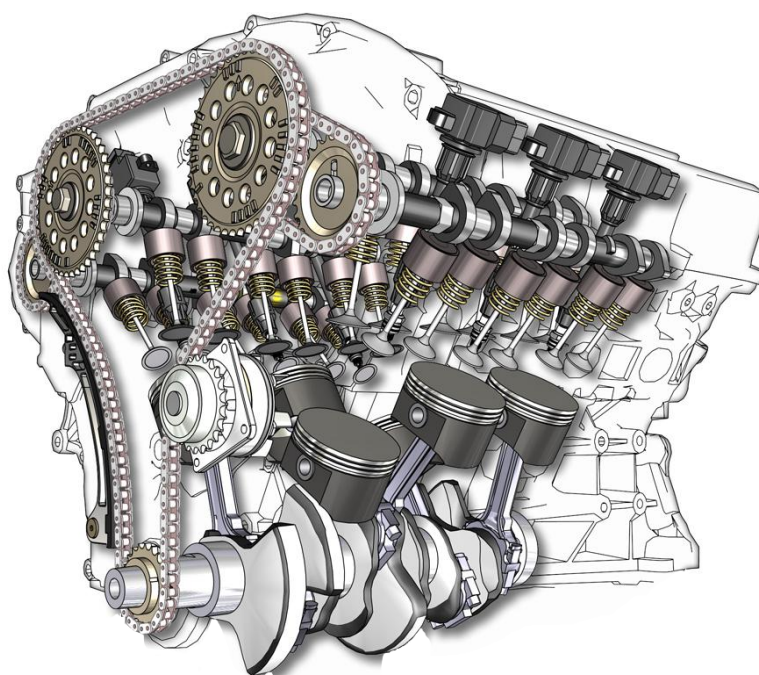
Next, in chapter 4, the experimental databases from Darmstadt and ECN used to validate the simulations of this work are presented. In chapter 5, the obtained results are shown and discussed, divided into sections related to each phase of the work (engine cold flow, fuel injection, spray combustion, full GDI engine simulation, and deep learning workflow). Finally, in chapter 6, a summary is provided with the conclusions of this work, followed by two appendices and one annex with further useful information.

## 2 A REVIEW ON INTERNAL COMBUSTION ENGINES RESEARCH

### 2.1 BASICS

The primary goal of a heat engine is to convert the chemical energy contained in a supplied fuel into useful mechanical energy, generating as few undesired outputs (such as heat and noise) as possible. This technology can be differentiated by its ways of operation, combustion methods, fuel type, working process, ignition method, mixture formation, and so on. The internal combustion engine, such as the one depicted in Figure 2, is in fact a member of a subclass of heat engines called reciprocating engines (HORROCKS, 2001). The members of this subclass generate mechanical power from the work done by a fluid over the surface of a piston confined within a cylinder. The translational movement of the piston is then transformed into rotational movement by means of a connecting rod and crank mechanism that drives the shaft.

Figure 2 – Illustration of a V6 internal combustion engine



Source: Wikimedia (2018c).

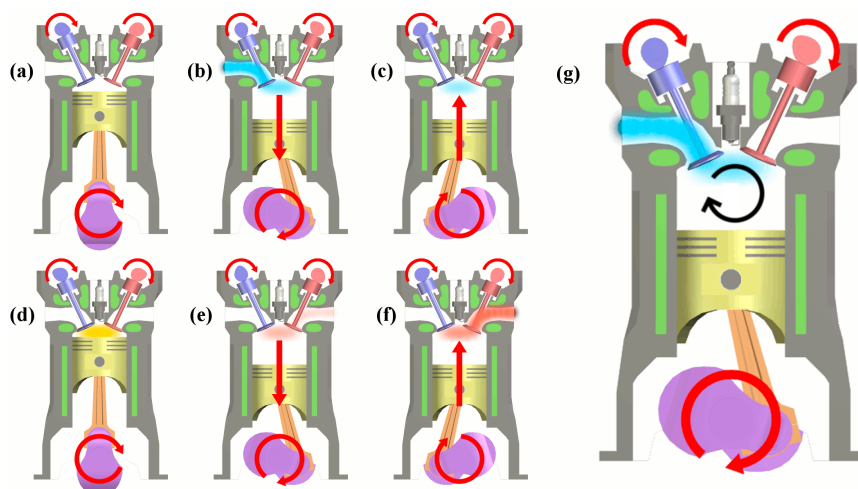
Moreover, the internal combustion engine is categorized by its ignition method, i.e. spark ignited such as gasoline engines or compression ignited like diesel engines. The focus of this work is on the technology combining the former method with the so called four-stroke cycle operation, successfully put in practice for the first time by the German Nikolaus Otto in the 1870s, although an unpublished patent attributed to the Frenchman Alphonse Beau de Rochas, issued in 1862, first described its working principle (HEYWOOD, 1988). The conventional four-stroke spark-ignited cycle is explained in Figure 3. At the beginning of the cycle (a), the piston is approaching top dead center (TDC) with the intake valves (left) starting to open and exhaust valves (right) finishing their closure. This is the only point in the cycle where both intake and exhaust valves are open, thus producing the so called "valve

overlap", during which fresh mixture is allowed from the intake to the exhaust ducts. This flow through the valves is beneficial since it helps to clean any residual fuel left in the combustion chamber as it cools down the exhaust valves. After that, the piston starts its movement downward to begin the **intake stroke** (b), as the exhaust valves completely close and the intake valves continue their downstream lift, allowing fresh mixture to enter the cylinder. This is the point of lowest temperature and pressure in the cycle, and also the moment during which the downward movement of the piston creates a pressure drop in the cylinder and induces the generation of turbulence. The large scales of this turbulent flow are known as the **tumble motion** (g), which is characterized by a large vortex around the plane along the intake-exhaust valves that is responsible for enhancing the mixture inside the cylinder. A few instants after the piston reaches its lowest position, here denominated as bottom dead center (BDC), the intake valves also close and the **compression stroke** (c) begins. With the air-fuel mixture trapped inside the cylinder, both temperature and pressure increase as the piston moves up. As the piston re-approaches the highest position, a **spark** (d) is created by a spark-plug located at the cylinder head, thus igniting the fuel. A flame then propagates from the spark position towards the piston and cylinder walls at a controlled speed, so the flame front reaches the piston just after it starts moving down again, generating work in the **power stroke** (e). Finally, as the piston reaches BDC and moves up one more time, the exhaust valves open to start the **exhaust stroke** (f), releasing the combusted charge into the exhaust pipes. At the end, the piston completes a second revolution as it approaches TDC to start a new cycle.

The fundamental difference between the engine cycle previously described and the engine cycle studied in this work is the presence of the additional event of fuel injection, that happens directly in the cylinder (sometimes referred to as combustion chamber), either during the intake or the compression stroke. The modeling of this event and all the related phenomena is detailed in chapter 3.

Now that an overview of the basic operation of the internal combustion engine has been given, a literature review on engine research methods can be addressed. In order to facilitate comprehension, this review is divided into three branches: an outlook on the early days of the technology, a review on experimental methods, and a review of computational methods.

Figure 3 – Schematic representation of the four stroke cycle

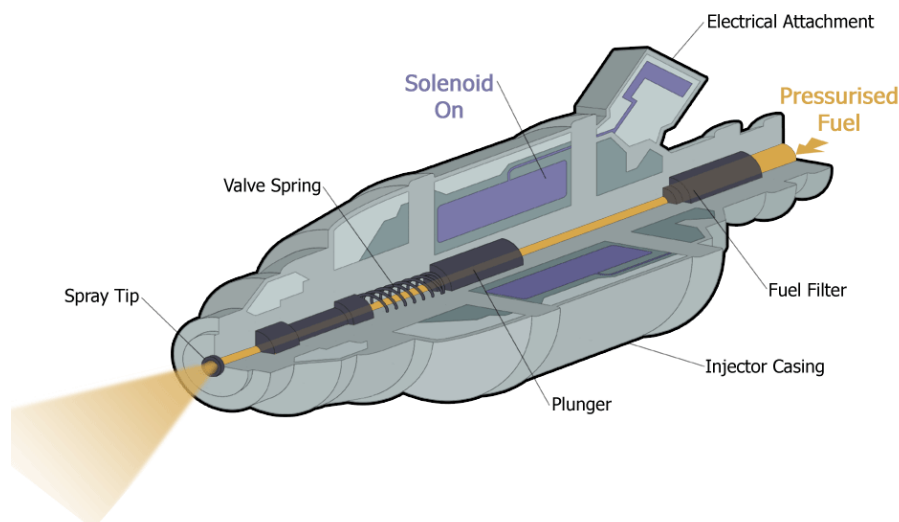


Source: Wikimedia (2018a).

## 2.2 EARLY DEVELOPMENT

Research on the four-stroke internal combustion engine can be traced back to the pioneer work of Otto in 1876. Although Beau de Rochas was the first to establish the principles of operation and to state the conditions for maximum efficiency of these engines, it was Otto who first built a working prototype and made it commercially viable (BASSHUYSEN; SCHÄFER, 2002). In 1879, another important historic figure of the engine industry, the German Karl Benz, combined the exhaust and intake processes with the power and compression strokes to successfully create the first two-stroke engine (SASS, 1962). Later, in 1883, two employees from Otto, Gottlieb Daimler and Wilhelm Maybach, were able to construct a four-stroke gasoline engine that achieved operational speeds of 600 rotations per minutes (rpm). This was an important milestone since the highest speed achieved by the first Otto engine was 160 rpm (HIERETH; PRENNINGER, 2007; DAIMLER, 2018; HEYWOOD, 1988). Only a few years later, in 1887, the British engineer James Atkinson patented his idea for an engine with expansion longer than the compression stroke (ATKINSON, ), using a complex crank mechanism. This approach brings some advantages under partial load operation (CHEN et al., 1998), which led some current engines to adopt a similar idea in modern designs in order to achieve fuel economy (KAWAMOTO et al., 2009). However, although thermal efficiency was directly correlated to expansion ratio, the bad quality of the gasoline back then limited compression ratios to less than four (HEYWOOD, 1988) in order to avoid engine knocking (early ignition). This limitation led to the invention of a new thermodynamic process for internal combustion engines by Rudolf Diesel (). Named after him, this new engine has the combustion process triggered simply by injecting diesel<sup>1</sup> directly into the combustion chamber (by means of a fuel injector such as the one depicted in Figure 4). Due to the direct injection of the fuel at the end of the compression stroke, higher compression ratios, and thus greater efficiency, can be achieved.

Figure 4 – Schematic diagram of a typical fuel injector



Source: Wikimedia (2018i).

<sup>1</sup> Fuel composed of around 75% saturated hydrocarbons and 25% aromatic hydrocarbons with average chemical formula  $C_{12}H_{24}$  (ATSDR, 2018; DATE, 2011).

In the 20th century, the events of the first and second world wars pushed the limits of many fields of science and technology, and the research on internal combustion engines was not an exception. The demand for both diesel and gasoline engines was high due to their extensive use in different applications, including for use in aircraft. However, since aircraft engines require good quality (high octane number) gasoline to operate properly, a better process for gasoline production was needed. This demand was not met until 1938, when the Frenchman Eugene Houdry proposed a catalytic process that allows high quality gasoline production in large scales (HOUDRY et al., 1938). This process was not only fundamental for the allies to win the war, but also enabled the gasoline engine industry to achieve much better levels of efficiency and performance (HEYWOOD, 1988).

Following the end of the second world war in the late 1940s, environmental problems caused by internal combustion engine emissions were already a concern. The issue became apparent when Prof. Haagen-Smit from The California Institute of Technology proved that the action of sunlight on organic compounds and nitrogen oxides in the atmosphere leads to a very toxic type of pollution (HAAGEN-SMIT, 1952). This and other works to follow in the field of environmental sciences led to the creation of legislation to control emission levels of internal combustion engines, first in the US, then in Europe, Japan and the rest of the world (NOBLE, 1955; SCHOETTLIN; LANDAU, 1961; SMITH; DAVIES, 1996). In response to that, the three way catalytic converter was introduced in the US in 1975, following in Japan and Europe in mid 1980s (ISLAM; HILDENBRAND; HOSSAIN, 2018), becoming a mandatory item soon after. This device was another invention by Eugene Houdry (), who in the 1950s formed a new company with the mission of reducing health risks associated with automotive emissions. His invention is able to convert unburned hydrocarbons and products of inefficient combustion such as carbon monoxide into less toxic pollutants.

In the 1970s, however, another historic event significantly impacted the automotive industry when the Organization of Arab Petroleum Exporting Countries (OPEC) proclaimed an oil embargo targeted at western countries, such as the United States. One motive for the embargo, among others, was the perceived support of those western countries for Israel during the Yom Kippur War (ISSAWI, 1978). With the rapid growth of crude oil prices, pressures to make internal combustion engines more efficient became more substantial, even after this diplomatic conflict was solved (HEYWOOD, 1988). In addition to that, growing concerns about the effect of  $CO_2$  emissions on climate change made the pressure for fuel economy even greater, with downsizing and turbo charging concepts getting special attention (MAROTEAUX, 2017; JANAS, 2017). Since then, huge efforts have been made to improve every aspect of the engine manufacturing process, from simply employing new materials aiming at weight and heat loss reduction to new concepts of design and operation. The direct injection spark ignition (DISI) technology is one of these attempts to improve efficiency, because the evaporative cooling effect of the fuel injection allows an increase in compression ratio without knocking becoming a problem (BAUMGARTEN, 2005). However, due to design differences, DISI engines may undergo problems such as misfire and poor burning cycles more often than conventional spark ignited engines. This is true because, in the case of DISI engines, the injected fuel has significantly less time to evaporate and form an ignitable gaseous mixture field in the vicinity of the spark-plug than in conventional engines, in which the fuel is already mixed within the gas at spark timing. Therefore, the large adoption



of gasoline direct injection engines for means of achieving better efficiency opened a very active research field, on which the following sections in this chapter will be focused.

## 2.3 EXPERIMENTAL METHODS

Conventional empirical methods have been used as tools by engineers and researchers since the early days of the internal combustion engine technology. After the discovery of the piezoelectric effect at the end of the nineteenth century, devices such as piezoelectric sensors have been successfully used for in-cylinder pressure measurements and injection timing control (HINCHET et al., 2018; JANAS et al., 2015; PETERSON et al., 2017a; BAUM et al., 2014). At the same time, thermocouples are commonly used for temperature measurements in experimental works, including the most recent ones (TORMOS et al., 2018; PORAN et al., 2018; MARR et al., 2010). However, in order to quantify most of the properties inside the engine in high resolution in both time and space, more advanced diagnostic techniques are necessary. Among these techniques, laser diagnostics methods, such as particle image velocimetry (PIV), planar laser induced fluorescence (PLIF), and Mie scattering, have been gaining popularity since the 1980s (ARCOUMANIS; WHITELAW, 1987).

These measurement techniques all bear in common a dependency on laser technology in order to operate. These require not only state-of-the-art apparatuses for their handling, but also well trained people for their operation and raw data processing. For instance, the PIV technique consists of seeding the fluid of interest with small particles to measure flow velocity vectors by emitting a planar laser at one end and recording the scattered light by the particles at the other end. The fundamental principle of this technique lies in the fact that those seed particles alter the light crossing their way and that the particles are small enough to follow the flow streamlines with very little influence over it (in other words, the Stokes number must be low). This altered light can then be captured by one or several cameras, so that after some post-processing, the information about the velocity field in a certain region can be reconstructed (RAFFEL et al., 2007). Baum et al. (2014) created an experimental work-bench to study the optical accessible Darmstadt engine, on which PIV techniques are employed in order to better understand characteristics of the engine air flow, direct injection, and combustion. The data produced by the group is then stored in a database to be used as a validation source for numerical simulations, which in conjunction with the experiments can be a powerful tool to improve the current understanding in the field. In Baum et al. (2013), the same group of experimentalists overcome the 2D limitation of PIV measurements employing tomographic PIV. In this particular PIV method, the thickness of the light sheet is expanded to a defined depth (perpendicular to the laser sheet plane). As a consequence, the light scattered by the trace particles illuminates a volume (in contrast to a plane in regular PIV), being collected by a set of several cameras (SCARANO, 2013). The importance of this method is clear since engine flows are highly turbulent, thus intrinsically three-dimensional. This is confirmed by one of the findings of their work, since they observed differences between 3D and 2D turbulent kinetic energy fields, especially during the intake stroke (BAUM et al., 2013). Moreover, other PIV works have been successfully applied to accurately measure the flow field in engines (RABAULT et al., 2016; BODE et al., 2017; KRISHNA; MALLIKARJUNA; KUMAR, 2016).

Apart from allowing accurate spatial and temporal single phase flow measurements, PIV techniques

can also be employed to obtain velocity fields of multiphase flows, including the case of fuel-sprays. Spicas et al. (2017b) used planar PIV to measure the velocity of the gaseous environment in a stationary combustion vessel used for experimenting fuel injection of the ECN “Spray G” test case. This test case aims at the study of an eight hole solenoid activated gasoline injector intended for use in GDI engines (ECN, 2018). In their work, they were able not only to provide reliable measurements of the gaseous phase velocity field in different sampling positions that numerical groups can use for validation, but also to study the mechanisms for inter-plume interaction and spray collapse. Although these experiments in stationary vessels help researchers to better understand the isolated events taking place during fuel injection, the behaviour of a spray injected in a stationary gaseous environment may differ substantially from sprays injected into the chamber of a GDI engine. With that in mind, Peterson et al. (2017a) applied tomographic PIV to study the effect of the fuel spray on the turbulence inside a DISI engine. It was found that the spray-induced turbulent enhancement remains for a significant time after end of injection (PETERSON et al., 2017a). In another study, Peterson et al. (2016) found that fuel injection can increase the engine turbulence levels up to 70%.

More than only assessing velocity fields, optical techniques can also be used to obtain information about the fluid temperature, as well as spray specific properties, such as droplet size distributions and spray penetration length. Labergue et al. (2010) obtained the droplet temperature information of a fuel-spray using laser induced fluorescence (LIF) imaging, another type of optical diagnostic in which the flow is seeded with a temperature-sensitive tracer. In the work of Parrish (2014b), spray penetration curves for both liquid and gas phases of the “Spray G” injection test cases were obtained using a high-speed imaging system based on schlieren and Mie scattering techniques (SCHULZ, 2010). Moreover, for the same injector, Parrish (2014a) obtained measurements of droplet diameter and droplet velocity distributions at several times after start of injection. All this information is important for the development of models that predict the important physical phenomena in fuel-sprays, such as heat-transfer and evaporation, atomization, droplet breakup, droplet collision, and so on.

Finally, experimental methods can also be employed to quantify properties of reactive flames and other reactive flows, such as the one inside internal combustion engines under fired operation. Pastor et al. (2016) performed an experimental study on reactive sprays of n-heptane, iso-octane, and intermediate blends between these two. In their work, the schlieren technique was used to measure trace values of OH emissions in order to quantify ignition delay, which is defined as the period of time between start of fuel injection and start of ignition, and the flame lift-off length, i.e. the farthest upstream location from the injector nozzle of high temperature combustion. Furthermore, Pickett (2005) used chemiluminescence to study flow flame temperature limits of reactive diesel sprays. He has shown that if the gas temperature is greater than 1000 K, flames as cool as 1500 K can have high combustion efficiency, which is positive in the pursuit of  $NO_x$  emissions reduction. In another work, Skeen, Manin and Pickett (2015) investigated the low and high temperature ignition characteristics of a n-dodecane spray. It was shown that high-ignition temperature initially appears at the radial periphery of the jet, consuming low temperature radicals. Ultimately, Peterson, Reuss and Sick (2014) employed LIF and PIV techniques to study the correlation between spray core development and combustion instabilities in an optical DISI engine, showing that rare events of misfire can occur if the flame

kernel happens to be surrounded by fuel-lean mixtures in a particular cycle. Furthermore, Sjöberg and Reuss (2013) used high speed imaging to investigate the effect of very late (near TDC) injection in a spray-guided GDI engine. They have shown that late injection can be very promising for the reduction of NO and particulate matter emissions, while maintaining relatively high combustion stability.

## 2.4 COMPUTATIONAL METHODS

### 2.4.1 Conventional Methods: Computational Fluid Dynamics

The use of computers to study internal combustion engines was a little explored subject of research until the mid-1970s, mostly because of the lack of computational resources. However, as computers became faster very rapidly (as dictated by Moore's law<sup>2</sup>), it became possible to tackle the difficult problems associated with engine flows using computational methods. As pointed out by Westbrook et al. (2005), during the early days the interest came from American governmental research groups, which first identified the difficulties and the problems related to engine in-cylinder phenomena in events like the Workshop on the Numerical Simulation of Combustion for Application to Spark and Compression Ignition Engines held in 1975. After that, great advances were achieved when some of these interested scientists persuaded the American government and automotive companies like GM, Ford and Chrysler, that only a joint effort of government, universities and industry would be able to address these problems (WESTBROOK et al., 2005).

Perhaps the most successful outcome of that joint effort was the development of the first widespread internal combustion engine simulation tool: KIVA. The KIVA code was born at the Los Alamos National Laboratories in a project funded by the United States Department of Energy (DOE). The idea for this came in 1976, just a few years after the first world energy crisis, when during a meeting sponsored by the National Science Foundation participants were asked to give their opinions on how to make engines more fuel efficient. One researcher from Los Alamos realized that the reactive fluid dynamics code they developed could be adapted for that purpose and the first talks towards that goal started (AMSDEN; AMSDEN, 1993). The first version of the code could do calculations only in two dimensions and had fixed boundary conditions. A few years later, mesh motion capabilities were added to simulate things like the movement of the piston. Then naturally, three dimensions were incorporated to the code, as well as sophisticated fuel-spray and reaction kinetics models.

Meanwhile, the field of computational fluid dynamics (CFD) in general also evolved substantially with the development of computational sciences. Already in the 1950s the first works on models to describe fluids governed by the Navier-Stokes equations started to arise, primarily with the goal of developing nuclear weapons (HARLOW, 2004). These first CFD codes were very inefficient, though it did not take long until researchers realized the potentials of the field for a diverse range of new applications of interest in science and engineering. This brought novel methods to improve the capabilities of CFD in making accurate predictions, such as the finite volume method (FVM). In the FVM method, more than creating a discrete representation of the computational domain in the form of a computational grid (or mesh), and deriving finite-differences between grid points by

<sup>2</sup> The Moore's law states that the number of transistors (or the number of floating operations per seconds - flops) in a dense integrated circuit doubles approximately every two years.

means of a truncated Taylor series, non-overlapping volumes are defined around each grid point (PATANKAR, 1980). Each set of governing equations can then be integrated over the boundaries of all these discrete volumes, so that fluxes between adjacent cells can be calculated. In this way, one can ensure conservation of mass, momentum, energy, and any other property by imposing that the flux of a quantity  $\Phi$  entering a volume must be equal to the flux leaving the same volume. Thus, conservation for any cell, and consequently the entire domain, is guaranteed.

Moreover, very early CFD models had to deal with turbulence, since this is a characteristic normally present in flows of interest for engineering (like engine flows). The earliest turbulent models ever proposed were based on the Prandtl's mixing length hypothesis (TOLLMIEN et al., 1961), which considers that the properties of a chunk of fluid are conserved for a certain length until it gets completely mixed with its surroundings. Thus, a simple model for the turbulent viscosity based on one characteristic length can be derived. However, an accurate description of turbulence requires a full solution of turbulent scales in a flow, from the largest to the smallest. The larger scales control the transport and mixing processes of the flow and are strongly influenced by the geometry, whereas the small scales depend solely on the amount of energy that is being transferred from the larger ones, having thus a more universal character (POPE, 2011). However, the solution of all turbulent scales down to the smallest one, also known as the Kolmogorov (1941) scale, was infeasible for most applications during the early days because the necessary number of grid elements would exceed the available memory of the best computers at the time. In order to circumvent this limitation, a new class of turbulent methods called Reynolds-Averaged Navier-Stokes (RANS) was created. In RANS, only the time averaged properties of the flow are considered in the solution, completely neglecting the fluctuating part. For that to work, all the information regarding the transfer of energy from the large scales to small scales of turbulence needs to be modelled. In one of the most successful existing RANS models, this is done by the solution of two additional transport equations, one for the kinetic turbulent energy ( $k$ ), and the second for the dissipation of  $k$  ( $\epsilon$ ), therefore the name  $k$ - $\epsilon$  model (JONES; LAUNDER, 1972). As the turbulent energy is now completely modelled in contrast to being solely resolved by the Navier-Stokes equations as in direct numerical simulations (DNS), there is no longer any requirement for extreme mesh refinement, and computational resources can be largely saved. Similarly to the way that the  $k$ - $\epsilon$  model is used to predict a variety of turbulent flows, other variations of RANS models are also widely used by both industry and academia, such as the  $k$ - $\omega$  model by Wilcox (1988) and the SST model by Menter (1994).

For a long time, RANS has been the normal approach to simulate engine flows. Many of these studies have been conducted with RANS implementations into the KIVA code. The ability of the code to deliver high-quality simulations of engine flows, fuel injection, and chemical kinetics supported by experimental validation is described in several reports (AMSDEN; O'ROURKE; BUTTLER, 1989; AMSDEN, 1997; AMSDEN et al., 1985). Using KIVA, Baritaud, Duglos and Fusco (1996) performed simulations of a DISI engine in order to test pollutant formation models for the prediction of NO and CO emissions, obtaining good agreement with measured results. The code was then again employed in Hong and Tarng (1998) to investigate the turbulence scales of the flow inside an optical engine during motored operation, achieving good predictions of turbulence intensity. In more recent years, Lucchini,

D'Errico and Ettore (2011) present a RANS simulation of a fuel-spray in engine conditions with the open-source code OpenFOAM, obtaining great performance with the employment of the adaptive grid refinement technique. Furthermore, using a distinct approach that treats both gas and liquid phases as a continuum medium, Baldwin et al. (2016) use a RANS code to study the internal and near-nozzle flow of the "Spray G" injector. Finally, in the work of d'Adamo et al. (2017), a statistical model for the prediction of knock in spark ignited engines is implemented in a commercial code, showing the capability of RANS models in anticipating highly transient effects in engines if probability density function (PDF) based models are employed.

Many of the solvers used in the publications discussed so far deal with gasoline direct injection engines. These were only possible due to the development of fuel-spray models that proved their validity after extensive comparison against experimental results. For instance, Reitz (1987) developed a droplet breakup model that works with the assumption that atomization and primary breakup are indistinguishable processes, proposing two mechanisms of secondary breakup, one for low and the other for high Weber number regimes. In other work, Beale and Reitz (1999) developed another breakup model in which droplets break based on the Kelvin-Helmholtz and the Rayleigh-Taylor instability models. Regarding particle collisions, O'Rourke (1981) came up with a stochastic method to predict collisions between fuel-sprays droplets that became the standard approach in Lagrangian based engine solvers. However, this model may become infeasible in some situations since its cost increases quadratically with the number of parcels (groups of drops) simulated. To overcome that problem, Schmidt and Rutland (2000) derived a new collision model from the probability equation of collision between two drops in a fixed volume, obtaining drastic improvements in both performance and accuracy. Furthermore, Zuo, Gomes and Rutland (2000) proposed an evaporation model capable of evaluating the evaporation of droplets taking into consideration the effect of superheat vaporization. Finally, computational methods have also been extensively used for the development of reaction mechanisms, so that simulations of DISI engines under fired operation can be achieved. These mechanisms can be extremely simple as a single-reaction mechanism (WESTBROOK; DRYER, 1981), moderately complex as in reduced mechanisms (ZHOU et al., 2016; WANG; YAO; REITZ, 2013), or very complex as in detailed mechanisms (WESTBROOK et al., 2011).

With further evolution of computational resources after the 2000s, a new framework for engine simulations known as large-eddy simulation (LES) not only became possible, but also preferable in the simulation of engines. LES can be seen as a compromise between DNS and RANS, in which the turbulent scales are filtered between large eddies and small eddies. The large eddies are any structures possible to be captured by the computational grid. Those larger vortices carry most of the total kinetic energy (at least 70%) and are normally very hard to model. In contrast, the sub-grid scales are not possible to be solved unless a much finer mesh is constructed, but for its isotropic character they can be easily modelled. Therefore, in LES the large scales of turbulence are resolved directly by the Navier-Stokes equations and the effect of the small vortices is added afterwards by the use of sub-grid models. The first ever LES model developed, the Smagorinsky (1963) model, makes the assumption that the small sub-grid scales are simply dissipated and, consequently, cause an increase in viscosity, using a single constant to model this viscosity increment. More sophisticated models have

been proposed using a more dynamic approach to determine the effect of the small scales on the flow, as seen in Germano (1992) and Kim and Menon (1995), but the Smagorinsky model still remains as one of the most used LES methods.

Richard et al. (2007) suggest that CFD modelling of internal engine flows should move towards the LES approach due to its ability to capture cycle-to-cycle variations, pioneering the use of this technique in an engine with good predictions backed by experimental data. Moreover, Vermorel et al. (2009)'s first attempt to study cyclic combustion variations observed experimentally, running nine consecutive LES cycles of a four-stroke spark-ignited engine with the AVBP code (AVBP, 2018). They point out, however, that the number of cycles computed may still be small to accurately draw statistical conclusions. In another study, Janas (2017) used the standard Smagorinsky model to simulate an optical engine with a code based on OpenFOAM, investigating the behaviour of the large vortices during the intake/compression stroke to determine the physics of the tumble breakdown at the end of the compression stroke. In a similar way for the same engine, Nguyen (2017) used advanced LES models implemented into the in-house PsiPhi code to investigate the amount of energy contained in the smallest turbulent scales and their impact on the combustion model when different numerical schemes are used. Finally, Vuorinen et al. (2011) shows that LES can also be successfully used for predicting fuel-spray flows using a Lagrangian-Eulerian approach to study the effect of droplet size in the mixture formation, improving the understanding in spray formation and suggesting a new correlation for the spray penetration length (VUORINEN et al., 2010b; VUORINEN et al., 2010a).

Ultimately, in recent years the CFD community has seen an increase in LES engine simulations, with some of these works being performed in supercomputers around the world. The outstanding performance of some of these facilities opened the possibility for LES simulations to achieve such high resolutions that it became clear that a new milestone was about to be achieved. This milestone was the employment of the DNS approach for the simulation of internal combustion engines, as seen in Schmitt (2014) and Mandanis et al. (2018), for the simulation of the flow in a simplified engine geometry. That is also possible now for more realistic engine flows, such as the optical engine from the University of Michigan (UMICH, 2018), using resources from the Argonne National Laboratories (ANL, 2018), as presented in Giannakopoulos et al. (2017). Although these DNS simulations are still a work in progress and only a few publications are available with their findings, some of their preliminary results show a level of detail never witnessed before in the field of engine research, paving the path for the next generation of state-of-the-art research on internal combustion engines.

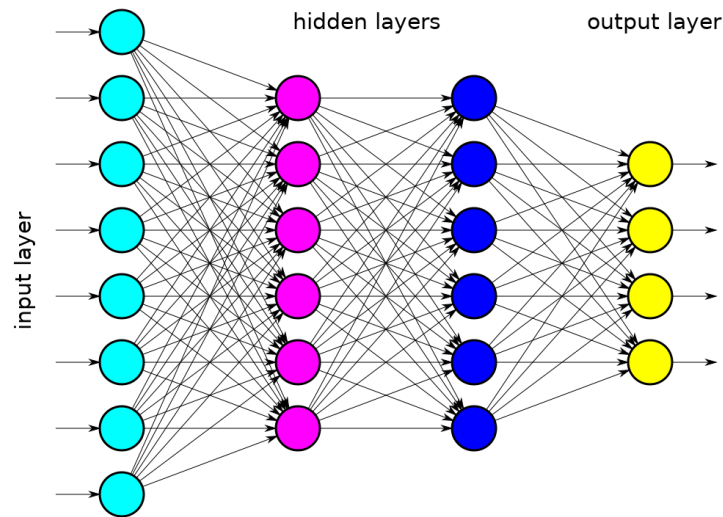
#### **2.4.2 Novel Methods: Artificial Intelligence**

In recent years, novel methods have been proposed to model fluid flows using technologies that are intended to be alternatives to CFD. Among these methods, one can highlight machine learning, which can be defined as programming a computer to be able to learn a certain task by feeding it lots of data related to that particular task (NG, 2017). Therefore, machine learning is nothing like conventional programming languages used to create CFD codes, such as C/C++ and Fortran, which are designed to enable the creation of programs that explicitly tell the computer what to do in a very consistent and systematic manner. In other words, given a certain input, one can precisely determine

the output of a conventional program by applying the set of tasks defined by its algorithm. In the case of machine learning, however, there is not a clear algorithm, but a procedure for the code to dig into a huge dataset, retrieving patterns and usable information from it. One way of doing this is using an artificial neural network (ANN), illustrated in Figure 5, a technology developed to mimic the behavior of the human brain, consisting of a net of interconnected computation nodes (neurons) that take some information as input (such as the raw data from which patterns are intended to be retrieved), perform some computation with this information, and pass it to a next layer of neurons, until a final layer that outputs the calculation performed by the net (NG, 2017). When a neural network consists of a large number of layers between the first layers (the one that receives the first input or raw data) and the last layer (the one that produces the output), it can be denoted as a deep neural network (DNN) (GUO; LI; IORIO, 2016; NG, 2017). Furthermore, if certain procedures and algorithms are applied to DNNs after feeding it large amounts of data, the network may learn how to make predictions based on the information it has been fed. For example, if a DNN is fed with data containing information about the housing market, such as the prices of houses in Guaratinguetá together with their important features (size in square meters, number of rooms, age, etc.), the DNN may learn how to make predictions of house prices in Guaratinguetá given a certain set of new features. This type of approach is known as supervised learning and such a DNN is known as a regression DNN. It has been proven very useful in the task of predicting flow properties normally computed with CFD, by either mimicking simulation data or modeling the partial differential equations used to generate CFD simulations (RAISSI; PERDIKARIS; KARNIADAKIS, 2019; RAISSI et al., 2018). In short, the outputs of the network are compared to the ground truth label (CFD data), and a certain metric is used to measure the deviation of the network output from that label, known as “cost”. Then, an iterative procedure for cost minimization is executed until a certain error threshold is achieved. For more comprehensive information on how neural networks can achieve these tasks, the reader can refer to classic architectures that have proven good predictive performance for several applications, such as the LeNet (LECUN et al., 1998), AlexNet (KRIZHEVSKY; SUTSKEVER; HINTON, 2016), GoogLeNet (SZEGEDY et al., 2015), inception nets (LIN; CHEN; YAN, 2014), residual nets (HE et al., 2016), and the VGG-16 net (SIMONYAN; ZISSERMAN, 2015).

Guo, Li and Iorio (2016) have employed a convolutional neural network (CNN) to make flow predictions of non-uniform 2D and 3D laminar flows. A CNN is a type of DNN based on the mathematical operation that gives its name and it is recommended for the task of pattern recognition from 2D or 3D images. Each convolutional filter, which is a parameter that can be learned from the training data, can recognize a different pattern, such as vertical lines, horizontal lines or any other structure hidden in the data. After feeding the CNN with CFD simulation data, the authors have employed it to make useful flow predictions for the same test-cases simulated with CFD, as shown in Figure 6, getting results very similar to the CFD simulation for simple flow configurations. In another study, Tompson et al. (2017) employ a different approach that intends to use CNNs in conjunction with CFD techniques instead of replacing them completely, leveraging the predictive power of deep neural networks with the precision and reliability of standard CFD solvers to obtain fast and trust worthy simulations. With their work, the authors were able to reduce simulation time by almost one order

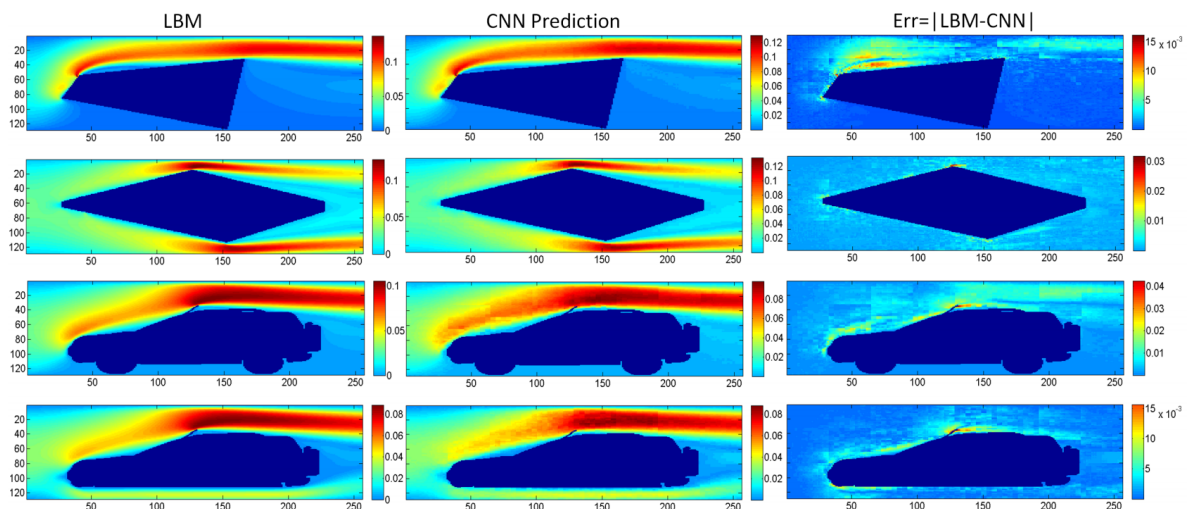
Figure 5 – Example of an artificial neural network with two hidden layers



Source: Wikimedia (2018e).

of magnitude, maintaining the same level of accuracy with increased stability. Differently from the data-driven approach previously introduced, one can also use a physics-driven approach, i.e., employ deep learning techniques to imitate the behavior of the Navier-Stokes equations themselves. Raissi, Perdikaris and Karniadakis (2019) introduce a supervised learning framework to train neural networks to respect any given laws of physics modeled by general nonlinear partial differential equations. The performance of the proposed work is tested and validated through a set of classical problems in fluids and other areas, such as quantum mechanics and reaction–diffusion systems. Moreover, Tompson et al. (2017) proposes a neural network to model the incompressible Navier-Stokes equations coupled with the structure’s dynamic motion equation to simulate flows around bluff bodies. This approach is able to accurately reconstruct velocity and pressure fields, as well as other structural parameters, given scattered data in space and time on the velocity field and the structure’s motion.

Figure 6 – Example of a CNN making flow predictions after learning from CFD simulation data



Source: Guo, Li and Iorio (2016).

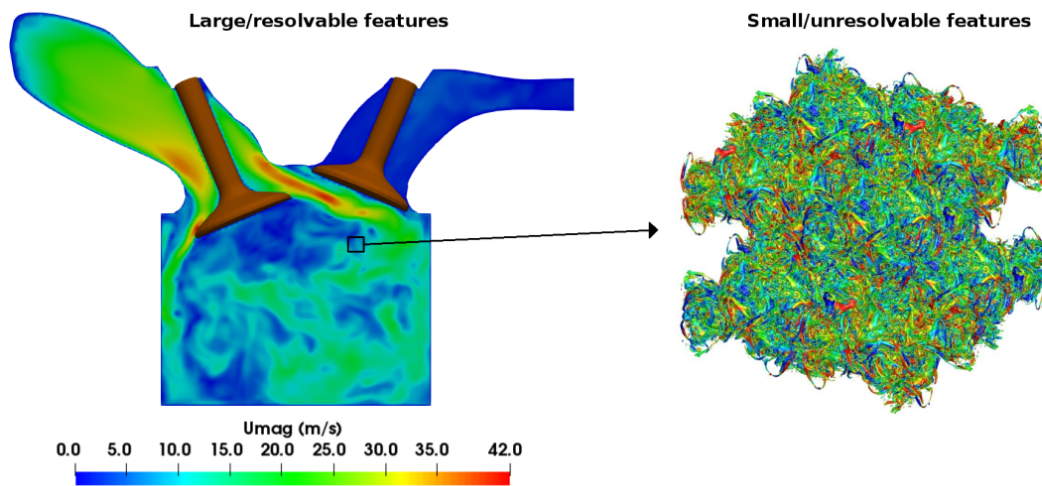


As previously discussed, CFD simulations of ICEs are a great challenge because they include several complex phenomena such as turbulence, multi-phase flow, combustion, and moving boundaries. Because of the broad discrepancies in length and time scales, the computations are both expensive and prone to neglect important information that is impossible to resolve in conventional ways. For example, high-fidelity turbulence models such as LES impose strict resolution requirements and highly accurate combustion kinetics models include a large number of species and reactions, driving up the cost of the calculations.

Meanwhile, deep learning can be a good alternative or aid to ICE CFD computations. While large-scale features of the simulations are specific to the given engine and fuel injection system, the small scales are expected to be universal. The physics of droplet breakup and combustion, for example, are not expected to vary from engine to engine. One expects that deep learning could capture the more universal features of small-scale engine physics (Figure 7) with better accuracy and less expense than traditional approaches.

The phenomena in an engine can be divided into (1) large, resolvable features such as tumble, swirl, fuel spray plumes, and heat release in contrast to (2) unresolvable features such as small-scale turbulence, droplets, and short-lived intermediate combustion species. The former category, consisting of the resolvable fluid flow, spray evolution, and major species transport, is specific to the engine configuration and must be simulated using traditional conservation equations. The latter features are better suited to simulation by a well-trained net. Thus, one can use the potential of deep learning presented coupled with a CFD approach to capture the two categories of phenomena.

Figure 7 – An ICE simulation showing the multi-scale nature of the problem



Source: Left figure (Elaborated by the author). Right figure (ATAK, 2019).

As all past and current methods for investigating ICE internal flows were here presented, this literature review chapter can thereby be concluded. From this review, both high-fidelity turbulence frameworks, such as LES and DNS, and the increasing development of artificial intelligence in the solution of practical problems in fluid dynamics can be highlighted as potential candidates for the prevailing frontier of knowledge in this field. Hence, the next chapter presents the modeling approaches used in this work in order to explore this frontier and make further contributions.

### 3 MODELING APPROACHES

In this chapter, the modeling approaches used in this work will be described. First, the LES governing equations followed by specific information on the turbulence modeling are presented. Finally, both conventional and novel engine modeling approaches are described.

#### 3.1 GOVERNING EQUATIONS

The LES filtered governing equations for the balance of mass (1), momentum (2), energy (3), and species (4) read:

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial (\bar{\rho} \tilde{u}_i)}{\partial x_i} = S_\rho \quad (1)$$

$$\frac{\partial (\bar{\rho} \tilde{u}_i)}{\partial t} + \frac{\partial (\bar{\rho} \tilde{u}_i \tilde{u}_j)}{\partial x_j} = \frac{\partial}{\partial x_j} \left[ \bar{\rho} \tilde{\nu} \left( \frac{\partial \tilde{u}_j}{\partial x_i} + \frac{\partial \tilde{u}_i}{\partial x_j} \right) - \frac{2}{3} \bar{\rho} \tilde{\nu} \frac{\partial \tilde{u}_k}{\partial x_k} \delta_{ij} - \bar{\rho} \tau_{ij}^{sgs} \right] - \frac{\partial \bar{p}}{\partial x_i} + \bar{\rho} g_i + S_u \quad (2)$$

$$\frac{\partial (\bar{\rho} \tilde{h})}{\partial t} + \frac{\partial (\bar{\rho} \tilde{h} \tilde{u}_i)}{\partial x_i} + \frac{\partial (\bar{\rho} \tilde{K})}{\partial t} + \frac{\partial (\bar{\rho} \tilde{K} \tilde{u}_i)}{\partial x_i} = \frac{\partial}{\partial x_i} \left( \alpha_{eff} \frac{\partial \tilde{h}}{\partial x_i} \right) - \frac{\partial}{\partial x_i} (\tau_{ij} \tilde{u}_j) + \frac{\partial \bar{p}}{\partial t} + \bar{\rho} g_i \tilde{u}_i + S_h \quad (3)$$

$$\frac{\partial (\bar{\rho} \tilde{Y}_i)}{\partial t} + \frac{\partial (\bar{\rho} \tilde{Y}_i \tilde{u}_j)}{\partial x_j} = \frac{\partial}{\partial x_j} \left( \mu_{eff} \frac{\partial \tilde{Y}_i}{\partial x_j} \right) + S_{Y_i} \quad (4)$$

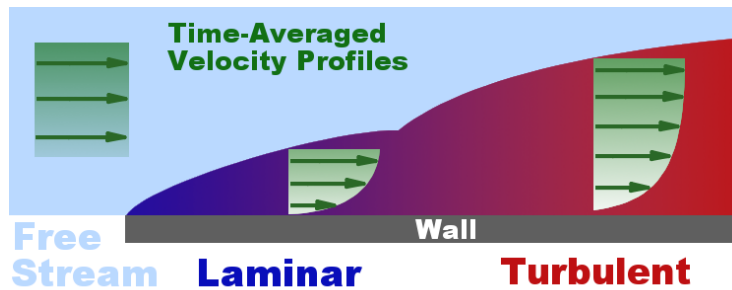
In the previous equations,  $u$  represents the velocity,  $\rho$  is the fluid density,  $\nu$  is the dynamic viscosity,  $Y_i$  is the mixture fraction of species  $i$ , and  $g$  is the acceleration of gravity. The energy equation is solved for the sensible enthalpy  $h$ ,  $\alpha_{eff}$  is the effective thermal diffusivity, and  $K$  the kinetic energy. The symbols  $S_\rho$ ,  $S_u$ ,  $S_h$  and  $S_{Y_i}$  are source terms added to represent the multiphase flow caused by the fuel injection into the engine cylinder (in case of no fuel injection, all these terms are equal to zero), and  $\mu_{eff}$  stands for the effective dynamic viscosity. The tilde over quantities implies that these equations are operating over the filtered quantities of the fluid of interest and the bar sign means averaging. Furthermore, the sub-grid stresses of the momentum equation,  $\tau_{ij}^{sgs}$ , are modeled using the standard Smagorinsky model ( $C_s = 0.168$ , as seen in Pope (2011)). For the derivation and solution procedure of these equations, please refer to Appendix A.

#### 3.2 TURBULENCE MODELING

The engine internal flow can be characterized as a conjunction of many different phenomena, from which turbulence may be the most challenging to study using computational tools. A common feature in flows of interest in engineering, turbulence can be defined as a random and chaotic three-dimensional flow, containing structures of varying size and behaviour. In the case of an engine, as air flows in the

intake pipes, a velocity boundary layer alongside the cylindrical walls is formed due to the action of viscosity, which means that fluid particles close to the walls stick to the pipe surface and are stationary, whereas the fluid away from the walls is free to flow with increasing velocity, as illustrated by Figure 8. Further away from the boundary layer, in the region known as free stream, flow particles are weakly influenced by the walls but may be strongly influenced by boundary conditions and neighboring flow currents. When these currents are composed of fluid particles flowing in smooth adjacent layers, the stream can be categorized as a well behaved laminar flow. As predicted by linear hydrodynamic instability theory, at a particular value of Reynolds number (ratio of inertial to viscous forces or  $Re = \frac{vd}{\nu}$ ) this well behaved laminar flow will transition to a chaotic turbulent flow, which is normally generated by growing instabilities associated with sheared flows (VERSTEEG; MALALASEKERA, 2007). This turbulence motion can then be intensified by the fast movement of valves and piston, and the high speed jet induced by fuel injection.

Figure 8 – Velocity boundary layer over a plate and transition from laminar to turbulent

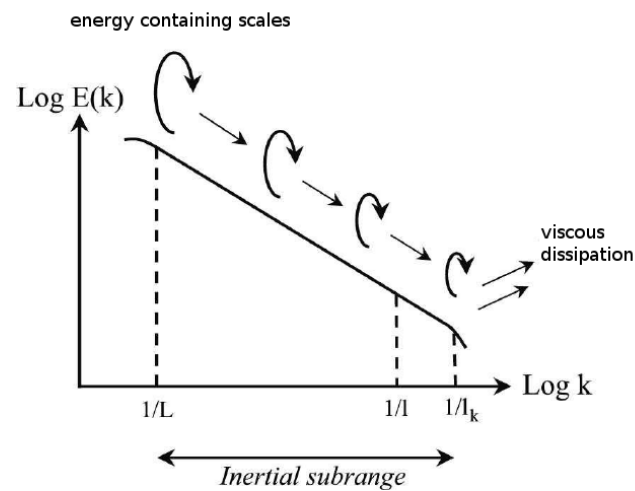


Source: Wikimedia (2018b).

Although described as disordered and irregular, turbulent flows present some well known properties that can be used to make reasonably accurate predictions of their behaviour. One of the properties is its random nature, being thus convenient to define flow variables in terms of the sum of a steady mean and its deviation, in a process known as Reynolds decomposition. Therefore, a turbulent flow can be characterized in terms of its mean velocity, which is strongly dependent on the physical constraints of the problem, and its fluctuations, which can be treated statistically. Furthermore, another essential characteristic of turbulence is the presence of rotational flows, also called vortices or eddies. Depending on the size of an eddy, different assumptions can be made. For example, on one hand, bigger eddies are highly influenced by inertial forces and the geometric constraints of the flow, controlling most of the transport and mixing processes. For this reason, these vortices are normally highly anisotropic, and therefore harder to be modelled, although not very high resolutions are necessary to resolve them directly with they Navier-Stokes equations. On the other hand, smaller vortices require very high resolution grids to be directly resolved, making their solution expensive in terms of computational resources. However, except when close to a wall, they are dependent only on the amount of energy being transferred from the larger scales, being thus assumed to have a more homogeneous and isotropic character for which statistical models can be derived. Hence, isotropy and homogeneity or lack of them are common properties used to describe turbulent flows, i.e. the more isotropic and homogeneous, such as the small scales, the easier it is to come up with modelling solutions, whereas the more anisotropic and inhomogeneous, such as the large scales, the easier it is to obtain a direct solution. Finally, the last

property to be highlighted here concerns the amount of energy carried by turbulent scales and how energy is transferred from larger to smaller scales in a process known as energy cascade, depicted in Figure 9. The energy cascade starts with the formation of large eddies of characteristic size  $L$  comparable to geometry constraints (like the diameter of a pipe). As declared above, these eddies carry most of the turbulent energy and last for a long time until transferring their energy and being dissipated into smaller eddies in a continuous process that goes down to the smallest scales of turbulence, known as the Kolmogorov scales (KOLMOGOROV, 1941). At the Kolmogorov scale not much energy is left, and the energy cascade is ceased by the viscous dissipation of the remaining eddies into heat.

Figure 9 – Energy spectrum of turbulent cascade



Source: Wikimedia (2018d).

As briefly mentioned in chapter 2, the complete solution of all turbulent scales down to the smallest one is known as direct numerical simulation (DNS), and it requires the employment of hefty computational resources (normally computer clusters or even super computers) due to the extreme high resolutions necessary to capture the scales at the right end of the energy spectrum. Although prohibitive in the lack of extraordinary computational resources, DNS needs fewer models for its implementation, and in consequence better accuracy can be obtained. In contradiction to DNS, Reynolds Averaged Navier-Stokes (RANS) models introduce additional differential equations to represent the production and transport of the entire turbulent kinetic energy (TKE) spectrum and the rate of dissipation of TKE in smaller eddies or other forms of energy, such as heat. Moreover, since RANS models operate over time averaged quantities, turbulent fluctuations are completely neglected. For those reasons, the RANS approach needs only moderate mesh resolution in order to provide reliable and accurate solutions, at the price of revealing fewer details than its DNS counterpart. However, prudence should be used when defining the mesh resolution to solve a particular problem, since RANS solutions are cell size dependent. Because of that, it is always well-advised to perform a mesh refinement study prior to the execution of a RANS, considering the fact that RANS solutions converge after a certain resolution is achieved. Subsequently, a compromise between the accurate but too expensive DNS and the affordable but less accurate RANS simulations can be obtained with the assessment of the large-eddy simulation (LES) approach. In the LES approach, reasonably finer grids than those needed for RANS simulations are required in order to directly resolve the hard to model large turbulent scales (that respond for up to

80% of the turbulent kinetic energy), whereas the presumably homogeneous isotropic small eddies are simulated by means of simple sub-grid models of varying complexity (POPE, 2004). This is done by applying a filter to all flow properties, in order to separate the influence of the large scales from the smaller scales. The simplest filter used in LES simulations is the computational grid itself, thus called implicit filtering since any flow structure smaller than the grid size is filtered out. In contrast, explicit filtering techniques are also common in the literature (RITTLER; PROCH; KEMPF, 2015; OLBRICHT et al., 2012), assigning common procedures such as Gaussian and top-hat filters. The advantage of LES is evident, since it can capture significantly more details than RANS for directly solving the energy containing vortices, without paying the high computational price of DNS. However, contrary to RANS, LES solutions does not necessarily converge with grid refinement, thus making it harder to assess the quality of a LES simulation.

### 3.2.1 The Smagorinsky subgrid model

In this work, an LES methodology was adopted for the description of turbulence. The subgrid stresses  $\tau_{ij}^{sgs}$ , as shown in equation (5), are modelled according to the Boussinesq model (BOUSSINESQ, 1877), and the assumption by Smagorinsky (1963) that the smallest scales are isotropic. In this case, Prandtl's mixing length hypothesis can be used to derive the subgrid viscosity in terms of one characteristic length and one velocity scale (VERSTEEG; MALALASEKERA, 2007):

$$\tau_{ij}^{sgs} - \frac{1}{3}\tau_{kk}^{sgs}\delta_{ij} = -\mu_{sgs}S_{ij} \quad (5)$$

where  $S_{ij}$  is the rate-of-strain tensor, calculated as shown in equation (6), and the turbulent or eddy viscosity  $\mu_{sgs}$  is modelled as presented in (7):

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_j}{\partial x_i} + \frac{\partial \bar{u}_i}{\partial x_j}\right), \quad |\bar{S}| = \sqrt{2S_{ij}S_{ij}} \quad (6)$$

$$\mu_{sgs} = \rho(C_s\Delta)^2 |\bar{S}| \quad (7)$$

where  $\Delta$  is the filter width and  $C_s$  is the Smagorinsky constant.

In conclusion, the above approximation for the eddy viscosity assumes that changes in the resolved fields are slow, so that subgrid eddies can adjust themselves quickly to the rate-of-strain tensor. Thus, a closure based on a single constant is not universally true and the constant value may have to be adjusted. In this work the original Smagorinsky constant value of 0.168, as in Pope (2011), was chosen based on good results achieved in previous ICE simulation works (JANAS, 2017; RIBEIRO, 2018).

## 3.3 MODELING OF GASOLINE DIRECT INJECTION ENGINES

### 3.3.1 Moving Mesh Strategy

In order to account for the moving parts of the engine such as piston and valves, the equations presented so far must be solved on a moving grid. For this task, a Laplace equation is solved to

determine the velocity field of the internal grid elements ( $\tilde{u}_{cell}$ ), as shown in (8):

$$\frac{\partial}{\partial x_j} \left( \gamma \frac{\partial \tilde{u}_{cell}}{\partial x_j} \right) = 0 \quad (8)$$

where  $\gamma$  is the mesh diffusion coefficient, which is a function of cell distance to the next moving boundary (JASAK; TUKOVIĆ, 2006). After calculating the velocity field of the internal grid elements, their positions can then be derived, allowing the mesh to move. These movement creates deformation in the mesh, which can be fixed by periodically mapping the results from one mesh to another, using the mapping approach explained in Appendix B.1.

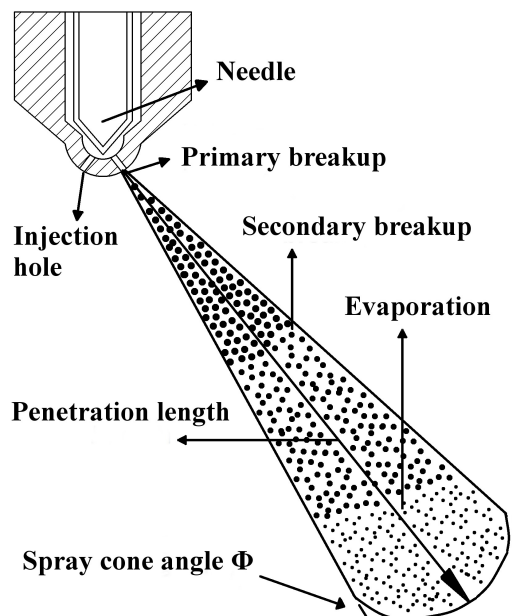
### 3.3.2 Fuel-Spray Modeling

For modeling the liquid phase, which is composed of many discrete tiny droplets or particles, Lagrangian methodology was employed, and the sub-models used to represent each phenomenon of the simulated full-cone spray (Figure 10) are described further. The parcel approach was used in this study to reduce computational effort. Thus, to circumvent the difficult problem of simulating each particle separately, a parcel is defined as a group of droplets with the same properties. In this way, only bins of particles need to be treated numerically. Another reasonable assumption is to consider that all particles are perfectly spherical (REITZ; BRACCO, 1986). The size distribution of the injected droplets is modeled by the Rosin-Rammler cumulative density function (CDF), as shown in (9):

$$F(x, n, d) = 1 - e^{-(x/d)^n} \quad (9)$$

where  $n$  is the constant of uniformity, and  $d$  is the characteristic particle size within the distribution (which depends on the fuel used).

Figure 10 – Schematic representation of a full-cone spray and the different phenomena undergone by a fuel-spray during injection



Source: Elaborated by the author.

The velocity magnitude of the injected parcels can be determined either by knowing the pressure drop between the injector nozzle and combustion chamber and applying the Bernoulli equation or by deriving it from the mass flow rate and injector geometry properties. Their directions are correlated with expressions that distribute the parcels over a predefined range, on which small particles tend to be injected within an angle (not greater than a maximum predefined value) and larger particles tend to go straight or stay within a small angle. When liquid fuel particles are injected at high velocities into the combustion chamber, they are decelerated by the gaseous environment. This exchange of momentum between the liquid and gas phases occurs due to the relative velocity between them. Equation (10), as shown in (VUORINEN et al., 2011), denotes the exchange of momentum by evaluating the drag force over a liquid parcel moving with relative velocity  $u_{res}$  with respect to the gas.

$$\frac{1}{6}\rho_P\pi d^3\frac{du_P}{dt} = \frac{1}{2}(u_g - u_P)|u_g - u_P|\rho_g C_D \frac{\pi d^2}{4} \quad (10)$$

The term on the left side in equation (10) represents the change in the acceleration of the liquid fuel particle, while the term on the right side denotes the drag force due to its relative motion to the gas phase. Subscript  $g$  refers to the gas phase and  $P$  refers to the liquid parcel of diameter  $d$ .

The drag coefficient  $C_D$  is calculated assuming spherical particles using the Reynolds number of the gas phase  $Re_g = d u_{rel} \rho_g / \mu_g$ . The correlation for evaluating  $C_D$  is presented in equation (11).

$$C_D = \frac{24}{Re_g} \left( 1 + \frac{1}{6} Re_g^{2/3} \right), Re \leq 1000 \quad (11)$$

$$0.424, Re > 1000$$

Finally, the parcels can be tracked by updating their positions according to equation (12).

$$\frac{dx_P}{dt} = u_P \quad (12)$$

As soon as the liquid particles get into the domain, they exchange not only momentum, but also thermal energy with the gas phase in the form of convective transport and radiation. The heat exchange increases the temperature until the particle reaches the boiling point and evaporates. This is very important for the mixing process, for combustion, and for the formation of pollutants, since the fuel does not start to burn until it is in the vapor phase.

Besides the assumption of perfect spherical particles, one may also neglect the effect of radiation, since it is small compared to that of convection (RANZ; MARSHALL, 1952). Furthermore, the evaporation modeling is based on the average flow field around the particles, because the calculation of the flow among them is not feasible. The energy balance for a liquid particle (BAUMGARTEN, 2005) is presented in equation (13):

$$m_P c_{p,l} \frac{dT_P}{dt} + \Delta h_{evap} \frac{dm_{evap}}{dt} = \lambda_g \pi d_P (T_\infty - T_P) \frac{\zeta}{e^\zeta - 1} Nu \quad (13)$$

The terms  $Nu$  and  $\zeta$  are shown in equation (14) and equation (15), respectively.

$$Nu = \frac{h d_P}{\lambda_g} \quad (14)$$

$$\zeta = \frac{\dot{m}_{evap} c_{p,vap}}{Nu \lambda_g \pi d_P} \quad (15)$$

In the previous equations,  $T_\infty$  is the temperature of the gas phase and  $T_P$  is the temperature of the parcel, while  $d_P$  is the parcel diameter,  $h$  is the heat transfer coefficient at the parcel surface,  $\lambda_g$  is the thermal conductivity of the gas mixture,  $Nu$  is the Nusselt number and  $\zeta$  is a dimensionless correction factor accounting for the reduced heat transfer due to the simultaneous mass transfer from the parcel to the gas phase. In equation (13), the first term represents the energy necessary to heat the liquid fuel parcel to a given temperature in an interval  $dt$ , where  $m_p$  is the mass of the parcel,  $T_P$  is its temperature, and  $c_{p,l}$  is the specific heat at a constant liquid fuel pressure. The second term accounts for the energy necessary to evaporate the parcel mass  $m_{evap}$  during the same time interval. The term on the right-hand side accounts for the heat transfer between the liquid and gas phase.

The appropriate Nusselt number that accounts for the relative velocity between the particle and gas is evaluated using the heat transfer correlation proposed by Ranz and Marshall (1952), as shown in equation (16):

$$Nu = 2.0 + 0.6Re^{1/2}Pr^{1/3} \quad (16)$$

in which the Reynolds number is denoted as  $Re$  and the Prandtl number is shown as  $Pr$ , both of which are calculated using the properties of the gas phase.

Finally, droplet vaporization is calculated as described by Zuo, Gomes and Rutland (2000). The model distinguishes between two kinds of vaporization: vaporization due to flash boiling (calculated with the experimental correlation proposed by Adachi et al. (1997)), and vaporization due to normal heat transfer with the gaseous phase, as described by equation (17) for the rate of vaporizing mass:

$$G_1 = \pi d Sh D_i \rho_g \ln \left( \frac{1 - Y}{1 - Y_s} \right) \quad (17)$$

in which  $d$  is the droplet diameter,  $Sh$  is the Sherwood number,  $D_i$  is the binary diffusivity coefficient,  $\rho_g$  is the gas bulk density, and  $Y$  and  $Y_s$  are the instantaneous vapour mass fractions in the cell and at droplet surface, respectively.

In the case of flash boiling conditions, the rate of vaporizing mass is giving by  $G_2$  in equation (18):

$$G_2 = 4\pi \frac{k}{c_p} r_0 \frac{1}{1 + G_f/G} \ln \left[ 1 + \left( 1 + \frac{G_f}{G} \right) \frac{h_\infty - h_b}{L(T_b)} \right] \quad (18)$$

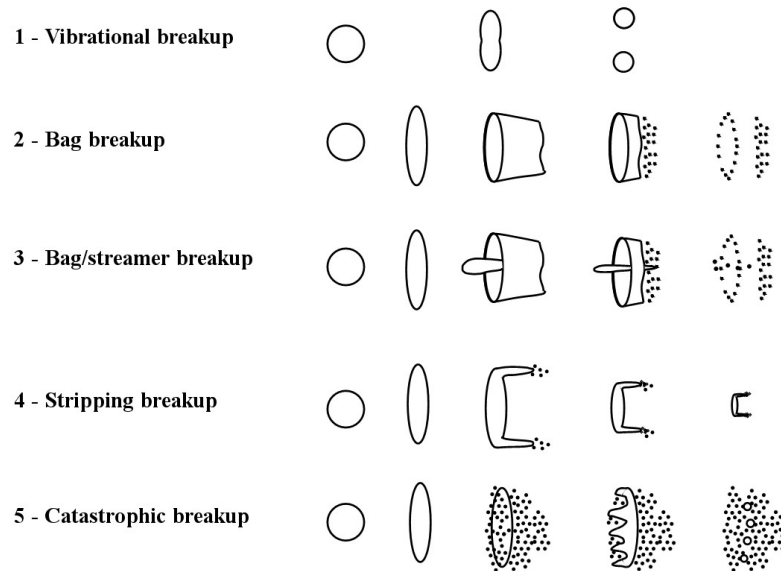
where  $G_f$  is the flash boiled vapor rate,  $k$  and  $c_p$  are the gaseous thermal conductivity and specific heat, respectively,  $r_0$  is the droplet radius,  $h_\infty$  and  $h_b$  are the gaseous enthalpy in the mixture and at the drop surface, respectively, and  $L(T_b)$  is the latent heat at the local boiling temperature.

In addition to the heat transfer and evaporation processes described above, fuel parcels are also subject to breakup as they travel along the domain. The formation of droplets from a round continuous liquid jet can be distinguished into four different breakup regimes, as shown by Reitz and Bracco (1986): the Rayleigh regime, the first and second wind-induced regimes, and the atomization regime. The formation of these first droplets is known as primary breakup and, in the case of fuel sprays,



atomization is the governing regime. These droplets are then subject to secondary breakup; i.e., new particles are formed from the collapse of previous ones, due to the high relative velocities in the liquid-gas interface. Baumgarten (2005) presents five different liquid drop breakup regimes: vibrational breakup (1), bag breakup (2), bag/streamer breakup (3), stripping breakup (4), and catastrophic breakup (5). These regimes are defined by Wierzba (1990) (see Figure 11).

Figure 11 – Droplet breakup regimes according to Wierzba (WIERZBA, 1990)



Source: Adapted from Wierzba (1990).

The relative velocity in the liquid-gas interface induces aerodynamic forces that create oscillations on the droplet surface. The increase in amplitude of these oscillations results in droplet breakup. The Weber number  $We_P = u_{rel}^2 D \rho_g / \sigma$  is the relevant parameter for the drop breakup mechanism. This number is a relation between inertial force (which tends to break up droplets into smaller particles) and surface tension (which tries to keep the parent droplets in their spherical shape). The symbol  $u_{rel}$  is the relative velocity between particles and gas,  $D$  is the particle diameter,  $\rho_g$  is gas density, and  $\sigma$  is droplet surface tension. As droplets become smaller, the surface tension necessary to break them increases, and therefore, the relative velocity must also increase to disintegrate the drop.

Experimental investigations have shown that distinct breakup regimes can be divided according to a range of Weber numbers. Table 1 presents the transition Weber numbers for different breakup regimes. Several models have been proposed in the literature for the numerical simulation of the secondary breakup process, from which the Reitz-Diwakar model was considered in this work (REITZ, 1987; REITZ; DIWAKAR, 1987).

The Reitz-Diwakar breakup model (REITZ; DIWAKAR, 1987) relies on two different assumptions. The first is that atomization and drop breakup near the nozzle within the spray cloud are considered indistinguishable; thus the blob method is used to represent the jet breakup into the first spray droplets. In this model, a detailed simulation of the phenomena near the nozzle is replaced by the injection of large spherical droplets, which are uniform in diameter and similar in size to the nozzle hole. In this work, as previously shown, a Rosin-Rammler distribution is used to assign diameter values to these first droplets, which are then subject to secondary breakup as they travel through the gaseous medium.

Table 1 – Transition Weber numbers for different regimes

Wierzba (1990)	We	Arcoumanis, Gavaises and French (1997)	We
1. Vibrational	$\approx 12$	1. Vibrational	$\approx 12$
2. Bag	$< 20$	2. Bag	$< 18$
3. Bag/streamer	$< 50$	3. Bag/streamer	$< 45$
4. Stripping	$< 100$	4. Chaotic breakup	$< 100$
5. Catastrophic	$> 100$	5. Sheet stripping	$< 350$
		6. Wave crest stripping	$< 1000$
		7. Catastrophic	$> 1000$

Source: Baumgarten (2005).

The second assumption is that secondary breakup can be distinguished between two droplet regimes: the bag breakup and the stripping breakup. Bag breakup will occur only if  $We_P$  is greater than a certain value, which is defined to be the bag constant (a value of  $C_{bag} = 1.5$  was assigned). For higher  $u_{rel}$ , stripping breakup may occur when  $We > C_{strip}\sqrt{Re}$ , where  $C_{strip} = 0.5$  is the stripping constant. Regardless of the breakup regime, the reduction in the particle radius is described by equation (19), where  $r_P$  is the particle radius and  $r_{stable}$  is the new stable particle radius after breakup.

$$\frac{dr_P}{dt} = \frac{-(r_P - r_{stable})}{\tau_{br}} \quad (19)$$

The characteristic breakup times  $\tau_{br}$  for bag and stripping breakup are given by equation (20), where  $C_1 = 0.785$  and  $C_2 = 10$  are constant models. The variables  $\rho_P$  and  $\rho_g$  represent liquid density and gas density, respectively.

$$\begin{aligned} \tau_{br-bag} &= C_1 \sqrt{\frac{\rho_P r_P^3}{2\sigma}} \\ \tau_{br-stripping} &= C_2 \frac{r}{u_{rel}} \sqrt{\frac{\rho_P}{\rho_g}} \end{aligned} \quad (20)$$

The new stable particle radius after breakup is given by equation (21) for both bag breakup and stripping breakup.

$$\begin{aligned} r_{stable-bag} &= \frac{6\sigma}{\rho_g u_{rel}^2} C_1 \\ r_{stable-stripping} &= \frac{\sigma^2}{2\rho_g^2 u_{rel}^3 \nu} \end{aligned} \quad (21)$$

### 3.3.3 Combustion modeling

As soon as fuel evaporates and mixes into the gaseous phase, combustion may start to take place given appropriate ambient conditions of temperature, pressure, and species concentrations are met. This phenomenon starts at the microscopic level with the occurrence of thousands of reactions among several chemical species. These reactions are normally activated in a chain process with different time scales and outcomes, following a certain reaction pathway that can be described according to reaction kinetics theory. Thus, assuming a chemical reaction between species  $A_1$  and  $A_2$ , resulting in products

$A_3$  and  $A_4$  of the form (22):

$$\nu_1 A_1 + \nu_2 A_2 = \nu_3 A_3 + \nu_4 A_4, \quad (22)$$

whereby  $\nu_i$  represents the so-called stoichiometric coefficients of the reaction and the equal sign means that the reaction can occur in both forward and reverse directions, the general equation for the reaction can be written as (23):

$$\sum_{i=1}^N \nu_i A_i = 0, \quad (23)$$

in which reactant coefficients are positive and products coefficients are negative.

In order to determine chemical equilibrium, i.e. condition in which forward and reverse reactions happen equally fast so that no macroscopic material conversion can be noticeable, the rate of change of concentration has to be evaluated for all species involved, as shown in equation (24) for a generic species  $A_i$ :

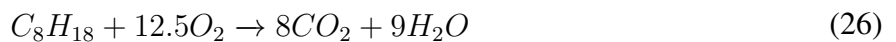
$$\frac{d[A_i]}{dt} = \nu_i (k_f [A_1]^{\nu_1} [A_2]^{\nu_2} - k_r [A_3]^{\nu_3} [A_4]^{\nu_4}) \quad (24)$$

where the first term on the right hand side represents the forward (f) reaction rate and the second term represents the reverse (r) reaction rate. The forward/reverse reaction rate coefficients  $k$ , strongly dependent on temperature, are calculated by the Arrhenius equation presented in Eq. (25):

$$k = AT^b \exp\left(-\frac{E_A}{RT}\right) \quad (25)$$

in which the pre-exponential factor  $A$ , the exponent  $b$ , and the activation energy  $E_A$  can be obtained from different sources (WESTBROOK; DRYER, 1981; MERKER et al., 2006; WARNATZ; MAAS; DIBBLE, 2006).  $T$  represents the absolute temperature and  $R$  the universal gas constant in SI units.

The aforementioned constant values were obtained empirically for the global oxidation reaction of a variety of fuels, such as iso-octane and n-heptane, in a single-step mechanism, i.e. fuel reacting with oxygen producing carbon dioxide and water vapour (WESTBROOK; DRYER, 1981), as shown in equation (26). These constants are summarized for a few selected fuels in Table 2. Moreover, the equations described above are handled numerically with the CHEMKIN chemistry package, which is designed to solve reaction kinetics mechanisms in chemically reacting flows (R. J. Kee and F. M. Rupley and J. A. Miller, 2013). More information about this setup can be found in the annex.



Overall, chemical reactions related to the oxidation of fuels are highly exothermic, which means that a great amount of energy is released during the formation of the combustion products. For the purpose of calculating the consumption/formation of species and the energy release into the domain due to combustion, the reaction rate must be evaluated and source terms  $\omega_i$  must be incorporated to both species and energy conservation equations. For instance, in the case of iso-octane combustion

Table 2 – Reaction rate parameters for a few selected fuels. Units are cm-s-mole-kcal-Kelvin

Fuel	A	$E_A$	a	b
$CH_4$	$8.3 \times 10^5$	30.0	-0.30	1.3
$C_6H_{14}$	$5.7 \times 10^{11}$	30.0	0.25	1.5
$C_7H_{16}$	$5.1 \times 10^{11}$	30.0	0.25	1.5
$C_8H_{18}$	$4.6 \times 10^{11}$	30.0	0.25	1.5

Source: Westbrook and Dryer (1981).

using the global single-step mechanism presented above, the species concentration source term for the fuel consumption can be written as in equation (27):

$$\omega_i = \frac{d[C_8H_{18}]}{dt} = -4.6 \times 10^{11} \exp\left(-\frac{30}{RT}\right) [C_8H_{18}]^{0.25} [O_2]^{1.5} \quad (27)$$

However, single-step global mechanisms are major simplifications of the real combustion processes which they are supposed to represent. Despite the fact these simplifications may be good enough to provide an overall picture of the chemical phenomena taking place in reactive flows, the downside is the negligence of a myriad of important species and reactions that may cause the combustion model to fail completely in giving any kind of useful prediction. In contrast to global approaches, detailed mechanisms describe the reaction kinetics in terms of thousands of reactions involving another thousands of species. For instance, Westbrook et al. (2011) proposed a detailed reaction mechanism for the oxidation of iso-octane that involves 1034 species in 4236 different reactions. Such a complete mechanism makes it possible to describe the entire chain of reactions that enable the combustion phenomenon to start, in a process known as ignition, and the prediction of numerous kinds of pollutants relevant to engine research, such as carbon and nitrogen oxides, as well as soot and other emissions. The ignition phenomenon is triggered by the formation of highly reactive radicals, such as hydroxyl (OH), from stable substances into the atmosphere where fuel is injected. In diesel engines, this process is known as auto-ignition since it starts simply by the increase of temperature and pressure during the compression stroke, and according to Baumgarten (2005), it involves four classes of reactions, namely chain initiation, chain propagation, chain branching, and chain termination. Regarding spark-ignited engines, the energy provided by the compression of the gas inside the cylinder is not enough to ignite the fuel, which is then triggered by the energy released by an electrical spark-plug. For the sake of simplicity, the spark is modelled here by assigning the adiabatic flame temperature of the fuel within a sphere in the vicinity of the spark-plug at spark time for a short period.

Although detailed mechanisms enable combustion models to give good predictions of phenomena related to reactive flows, the high computational cost necessary to solve all the reaction kinetics equations may become prohibitive, especially in the case of three-dimensional LES simulations. In order to reach a compromise between the computationally expensive detailed mechanism and the over simplified single-step mechanism, reduced mechanisms are developed. This is done in a process of simplification of a detailed mechanism by neglecting unimportant reactions and/or grouping several others in individual equations. Thus, computational resources are saved by decreasing the number of species and reactions, but still keeping a reasonable amount of detail, which considerably surpasses the modelling capability of global mechanisms. For instance, in this work the reaction mechanism

considering 43 species in 144 different reactions developed by Zhou et al. (2016) is employed. Table 3 shows a summary of some important reactions (complete mechanism is found in annex), with their respective Arrhenius coefficients. The choice for this mechanism is simultaneously justified by its simplicity and its capability to predict the phenomena of ignition delay, flame lift-off length, and flame speed reasonably well.

Table 3 – Some selected forward reactions of Zhou et al's mechanism

Reaction	A	b	$E_A$
$C_7H_{16} + O_2 \Rightarrow C_7H_{15} + HO_2$	$2.1 \times 10^{16}$	0	$4.6 \times 10^4$
$C_7H_{16} + OH \Rightarrow C_7H_{15} + H_2O$	$5 \times 10^{12}$	0	$3 \times 10^3$
$C_7H_{15} + O_2 \Rightarrow C_7H_{15}O_2$	$2.2 \times 10^{12}$	0	0
$C_7H_{15}O_2 \Rightarrow C_7H_{14}OOH$	$2.2 \times 10^{11}$	0	$1.9 \times 10^4$
$C_7H_{14}OOH + O_2 \Rightarrow O_2C_7H_{14}OOH$	$3.6 \times 10^{11}$	0	0
$O_2C_7H_{14}OOH \Rightarrow C7KET + OH$	$1.25 \times 10^{10}$	0	$1.7 \times 10^4$
$C7KET \Rightarrow C_5H_{11}CO + CH_2O + OH$	$9.98 \times 10^{15}$	0	$4.3 \times 10^4$
$C_8H_{18} + O_2 \Rightarrow C_8H_{17} + HO_2$	$8.5 \times 10^{16}$	0	$4.6 \times 10^4$
$C_8H_{18} + OH \Rightarrow C_8H_{17} + H_2O$	$1.5 \times 10^{12}$	0	$3 \times 10^3$
$C_8H_{17} + O_2 \Rightarrow C_8H_{17}O_2$	$2.1 \times 10^{12}$	0	0
$C_8H_{17}O_2 \Rightarrow C_8H_{16}OOH$	$2.1 \times 10^{11}$	0	$2.18 \times 10^4$
$C_8H_{16}OOH + O_2 \Rightarrow O_2C_8H_{16}OOH$	$2.58 \times 10^{11}$	0	0
$O_2C_8H_{16}OOH \Rightarrow C8KET + OH$	$3.9 \times 10^{10}$	0	$1.7 \times 10^4$
$C8KET \Rightarrow C_6H_{13}CO + CH_2O + OH$	$12.58 \times 10^{15}$	0	$4.3 \times 10^4$

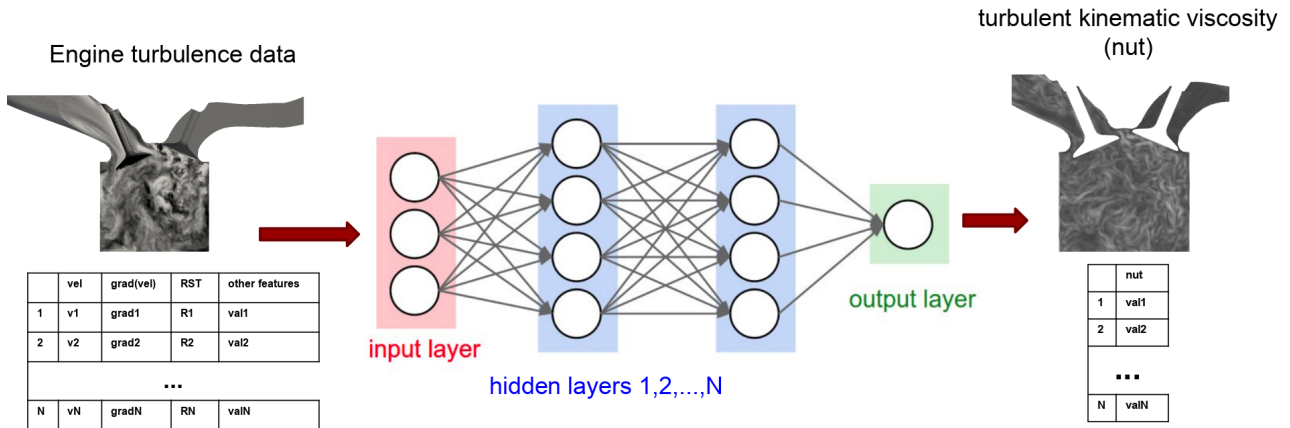
Source: Zhou et al. (2016).

In summary, the conjunction of all the sub-models described in this chapter are employed to simulate the multiphase reactive flow phenomena related to direct-injected internal combustion engines. In spite of the efforts of this work in using modelling approaches with good compromise between accuracy and computational limitations, many assumptions and constant tuning are necessary. Therefore, whenever experimental data are available, they will be used to assess the quality of the simulations by means of qualitative and quantitative comparisons.

### 3.3.4 OFICEnet

In order to leverage from the extensive data from ICE simulations produced using the conventional numerical models presented above, this work proposes to train an artificial neural network (ANN) with the generated CFD dataset to learn how to make predictions of small scale turbulence properties, such as the turbulent kinematic viscosity (Figure 12). As briefly explained in chapter 2, the reason for choosing the small scale turbulence behavior as modeling goal is the assumption that such scales are universal, thus easier to generalize than the large turbulence counterpart. To accomplish this task, the Keras framework (CHOLLET, 2015) is employed. Keras is an open-source python written neural network library which, running on top of TensorFlow deep learning framework, is able to combine the robustness and performance of a well established deep learning framework with an user-friendly platform designed to enable fast experimentation with neural networks (TENSORFLOW, 2019; CHOLLET, 2015).

Figure 12 – Schematic representation of OFICENet



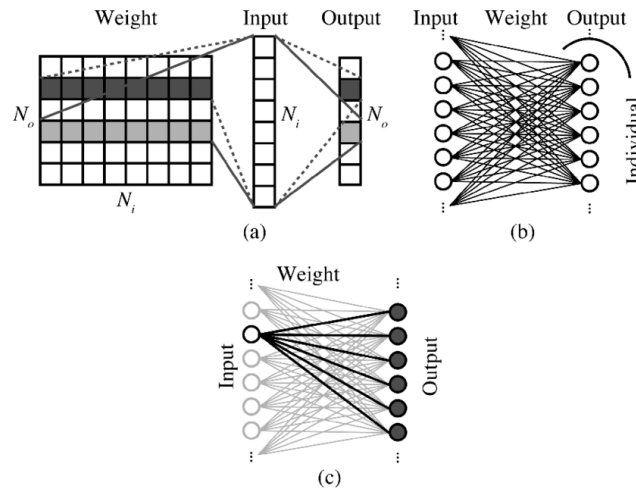
Source: Elaborated by the author.

The proposed neural network architecture, named OFICENet, is formed by fully connected (FC) layers of neurons, the most traditional configuration for a neural network. In the configuration used in this work, all inputs (velocity field and magnitude of the velocity gradient) are connected to all neurons of the first hidden layer, which are connected to the second hidden layer, and so on until the output layer, where a single output (turbulent kinematic viscosity) is calculated. A schematic representation of an FC layer is provided in Figure 13. As shown in the figure, the connection between the input layer and one training example represented by an input vector is given by a matrix multiplication between the first layer's weight matrix of dimensions  $N_o \times N_i$  and the input vector of dimension  $N_i \times 1$  (in which  $N_i$  corresponds to the number of features of each training example), thus producing an output vector of dimension  $N_o \times 1$ . This  $N_o \times 1$  output vector is then treated as input to the first hidden-layer of dimension  $N_o \times N_o$  for a certain number of times (corresponding to the number of layers between the input and the output, or the so called hidden-layers). For the calculation of the output, the output layer's  $1 \times N_o$  weight vector is multiplied by the  $N_o \times 1$  output vector given by the last hidden-layer. This process can be carried out for all training examples if an  $N_i \times N_e$  input matrix is fed to the first layer of the network, in which  $N_e$  corresponds to the number of training examples. This way, the output obtained will be a vector of dimensions  $1 \times N_e$ , which corresponds to one single output for each of the  $N_e$  training examples.

Furthermore, after each matrix multiplication described above, an activation function is calculated element-wise on each output vector. An activation function is necessary to limit the range of values outputted by a neuron. For instance, in the case of logistic regression (classification problem), it may be desirable to use a Sigmoid function as activation, in order to limit the outputs to just two possible values (0 or 1). In the case of the current problem (prediction of turbulent properties such as the turbulent kinematic viscosity), the activation has to produce continuous values. Thus, in this work the rectified linear unit, ReLU (NAIR; HINTON, 2010), is used as activation. Mathematically, it is defined as  $f(x) = \max(0, x)$ . This function was chosen instead of the standard linear activation (i.e. no activation function) because it induces sparsity, which means that neurons will not activate in all instances (it does nothing for negative inputs). This behavior is desirable because in sparse networks neurons are more likely to be activated only when identifying meaningful patterns, thus being less prone to over fitting,

and calculations in a sparse network are faster than in a dense network. For the last layer, however, no activation is used (linear activation), since this is a best practices recommendation for the case of regression networks (NG, 2017).

Figure 13 – Fully connected layers



Source: Ando et al. (2017).

The process of calculating an output value for each training example by consecutive matrix multiplications explained above is known as forward propagation. In the case of supervised learning, after calculating the output result (neural network prediction), its value must be compared against the ground truth result (CFD). The discrepancy between these values is obtained by a loss or cost function. In the case of discrete outputs (classification problem), the loss function can be given by means of the complement of the accuracy (i.e., the fraction of correct predictions made by the network). In the case of regression, a common metric for the loss function is the mean squared error (i.e., the sum of the squares of the difference divided by the number of training examples). After that, the calculated loss values are propagated back from the output layer to the input layer in a procedure known as back propagation (LINNAINMAA, 1970). In this procedure, the inverse operations performed during forward propagation are executed using the loss vector as input. In this way, it is possible to calculate the derivative of the values given by all neurons with respect to the number of iterations (number of times the information is propagated back and forth from the input to the output layer) for all layers of the network. The information about these derivatives at each iteration (also known as "Epoch") is then fed to an optimizer function, which intends to reduce the value of the loss function at every Epoch by updating the values of the weight matrices of the network in a procedure known as gradient descent, as shown in equation (28):

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \gamma \nabla F(\mathbf{a}_n) \quad (28)$$

The idea behind gradient descent is in the fact that a multi-variable function  $F(x)$ , such as an artificial neural network, when differentiable in the vicinity of a point "a" decreases fastest if one updates its value at "a" in the direction of the negative gradient of  $F$  at "a" ( $\nabla F(a)$ ). The pace of this update is given by a learning rate  $\gamma$ , which multiplies the gradient at each gradient descent iteration.

In the field of ANN, gradient descent is applied to update the weights of the network at every Epoch with the goal of minimizing the loss function. There are different variants of optimizers (i.e. variations of the gradient descent procedure) that one can use for this task. In this work, the Adaptive Moment Estimation, or Adam (KINGMA; BA, 2017), optimizer was employed. The idea of the Adam optimizer is not to simply set a constant learning rate for gradient descent, which if not wisely chosen can make the training process too long (too small learning rate) or it can make the optimizer to diverge (too big learning rate), but to include the idea of momentum to correct the learning rate. For this task, the momentum estimates of past and past squares gradients of  $F$  are given by  $m_t$  and  $v_t$  by equations (29):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t \quad \text{and} \quad v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \quad (29)$$

Since vectors  $m_t$  and  $v_t$  are initialized with zero values, the optimization procedure is initially biased towards zero. In order to counteract that, corrected momentum estimates are given by equations (30):

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t} \quad \text{and} \quad \hat{v}_t = \frac{v_t}{1 - \beta_2^t} \quad (30)$$

The Adam update rule for the weights at every iteration is then given by equation (31):

$$\mathbf{a}_{n+1} = \mathbf{a}_n - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t \quad (31)$$

for which the values suggested by the authors for the parameters are 0.9 for  $\beta_1$ , 0.999 for  $\beta_2$ , and  $10^{-8}$  for  $\epsilon$ . Although a step size  $\eta$  must still be chosen arbitrarily, the Adam authors have shown empirically that this method works well as an adaptive learning-method algorithm (KINGMA; BA, 2017).

Furthermore, an overview of the setup for the OFICENet is given in Table 4. The network is composed of 4 layers (3 hidden + output layer), with 20 neurons per hidden layer using the ReLU function (NAIR; HINTON, 2010) as activation. Finally, the training is performed using a mini-batch of size 100, which means that the network processes only 100 examples at time. This is necessary to not to “overwhelm” the network, since each engine CFD mesh has between 2 and 4 million cells.

Table 4 – Information on OFICENet architecture and solution strategy

Type of network	Fully-connected layers
Number of hidden layers	3
Number of neurons per layer	20
Type of activation	rectified linear unit (ReLU)
Number of features (inputs)	6 (U and gradUMag in x, y, and z)
Output	$\nu_t$ (turbulent kinematic viscosity)
Loss metric	mean absolute error
Batch size	100
Optimizer	Adam

Source: Elaborated by the author.

In order to illustrate how to design the OFICENet using the Keras framework, Frame 1 presents the algorithm necessary to create the model and to define the training operations of this network. Under the “create\_model” function, the fully connected layers (in Keras denoted as “Dense”) are defined. First, in line 5, the input layer, that will place the values of the features presented in the previous table, is



defined. After that, from line 8 to 10, the three hidden layers, each with 20 neurons, are created. Then, the output layer is defined in line 13. With exception of the output layers, all layers use the “ReLU” activation function. The “train()” function defines the parameters for training the net. Line 19 sets the optimizer and the loss metric. Finally, line 21 inputs the training data (X\_train, matrix placing feature values) with their respective labels (y\_train, vector placing turbulent kinematic viscosity values), the maximum number of iterations (also called “epochs”), the mini-batch size (100), and the validation split. The validation defines the percentage of training data examples that will be taken away from training (in this example a value of 0.2 or 20% is set) and used to test the network performance after training is complete. It is important to test the network with examples not seen during training to make sure that the network is not “overfitting”, i.e., being able to find a good fit for the training data, but not being able to generalize when new data is presented. In the general context of physical modeling, a model that “overfits” to a certain dataset is not able to make good predictions, because it is biased towards that specific dataset.

#### Frame 1 – OFICenet algorithm

---

```

1  def create_model():
2
3      model = Sequential()
4      // The Input Layer :
5      model.add(Dense(20, kernel_initializer='normal', input_dim = X_train.shape
6                      [1], activation='relu'))
7
8      // The Hidden Layers :
9      model.add(Dense(20, kernel_initializer='normal', activation='relu'))
10     model.add(Dense(20, kernel_initializer='normal', activation='relu'))
11
12     // The Output Layer :
13     model.add(Dense(1, kernel_initializer='normal', activation='linear'))
14     return model
15
16  def train():
17
18     // Compile the network :
19     model.compile(loss='mean_absolute_error', optimizer='adam', metrics=['
20                   mean_absolute_error'])
21
22     model.fit(X_train, y_train, epochs=MaxIter, batch_size=100,
23             validation_split = 0.2)

```

---

Source: Elaborated by the author.

This chapter introduced the conventional and novel modeling approaches proposed in this work. In order to test the conventional CFD approach, some test-cases together with experimental validation data are used to validate each individual part of the code, regarding the modeling of the engine internal flow, the fuel-spray, and the combustion phenomena. These databases and their creators are introduced in the next chapter. In order to validate the novel OFICenet modeling approach, comparisons of the OFICenet results to the ground truth CFD results will be presented, first against data seen during training (in other words, checking the ability of the model in overfitting), and second, trying to make a real prediction, i.e., calculating and comparing against results never seen during training.

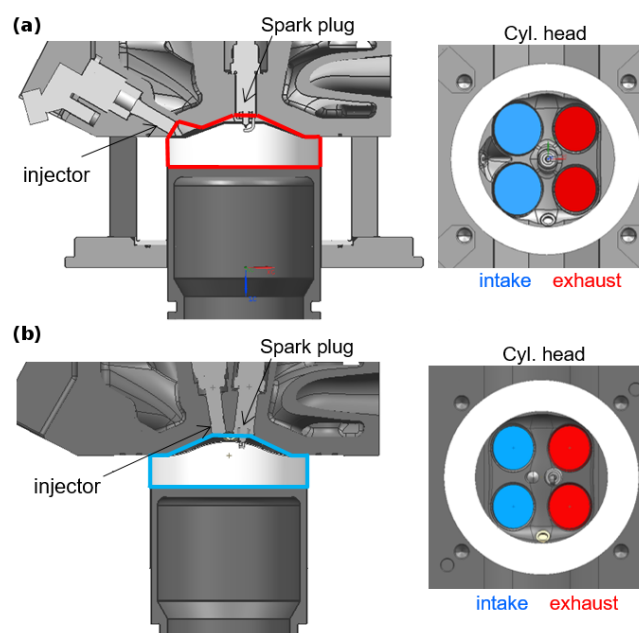
## 4 EXPERIMENTAL DATABASES FOR VALIDATION

The first database to be introduced is the Darmstadt optical engine used to validate the engine internal flow simulations exhibited in chapter 5 (section 1) and to serve as a test-case for the development of the GDI engine solver presented in chapter 5 (section 4). The experimental measurements of this database are credited to collaborators from TU Darmstadt (BAUM et al., 2014). Following that, the ECN spray injection test cases are introduced, discriminating between the non-reacting gasoline injection “Spray G” (chapter 5.2) and the reacting diesel/gasoline “Spray A” (chapter 5.3) cases. These experimental measurements are credited to different collaborators to the ECN (ECN, 2018).

### 4.1 OPTICALLY ACCESSIBLE INTERNAL COMBUSTION ENGINE

The Darmstadt engine, illustrated in Figure 14, is an optically accessible single-cylinder direct-injected four-stroke spark-ignited engine, located at the Technische Universität Darmstadt, in Darmstadt, Germany (FREUDENHAMMER et al., 2015; PETERSON et al., 2017a; BAUM et al., 2013; PETERSON et al., 2017b; PETERSON et al., 2016; PETERSON et al., 2013). The engine has four valves (two for the intake and two for the exhaust) mounted in a pent-roof cylinder head and can be operated either with port-fuel injection or with direct injection. For direct injection, two cylinder head variants are available. The first variant, the wall-guided (WG) cylinder head, is used when fuel is intended to be injected from the upper side of the cylinder, so that the spray cloud is guided to the spark-plug position by the piston wall. The second variant, the spray-guided (SG) cylinder head, is used for a distinct injection mode, in which the fuel spray is directly pointed towards the spark-plug (TU Darmstadt, 2018). In this work, the SG Darmstadt engine is simulated.

Figure 14 – The Darmstadt engine cylinder head variants. (a) Wall-guided and (b) spray-guided



Source: TU Darmstadt (2018)

Further information regarding the SG Darmstadt engine, as well as the operation conditions simulated in this work are found in Table 5:

Table 5 – Darmstadt engine, (BAUM et al., 2014)

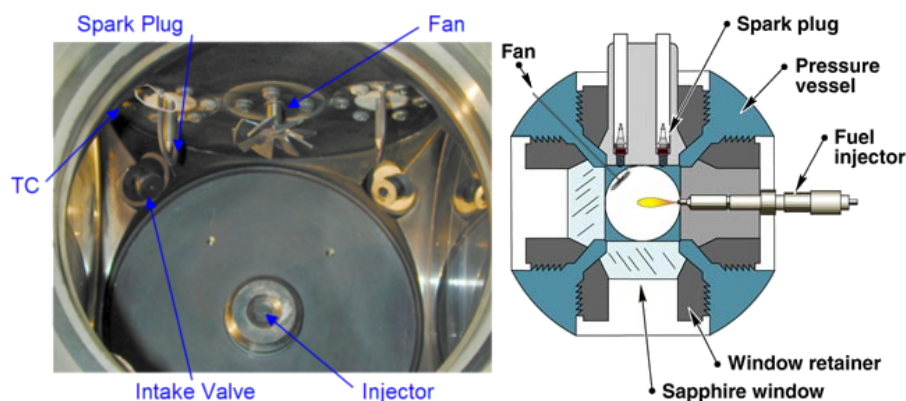
Number of cylinders	1
Bore	86 mm
Stroke	86 mm
Displacement	499.6 cm <sup>3</sup>
Number of valves	4 (pent-roof cylinder head)
Compression ratio	8.7:1
Engine speed	800 +/- 7 min <sup>-1</sup>
Intake pressure	0.95 bar
Intake temperature	295 K
Exhaust pressure	1 bar
Exhaust temperature	316.7 K
Intake valve open	35 CAD before TDC
Intake valve close	55 CAD after BDC
Exhaust valve open	75 CAD before BDC
Exhaust valve close	15 CAD after TDC

Source: TU Darmstadt (2018).

#### 4.2 OPTICALLY ACCESSIBLE COMBUSTION VESSEL

The common approach to experimentally study fuel sprays is the use of optically accessible combustion vessels, as the one depicted in Figure 15, in combination with state-of-the-art optical diagnostic techniques, such as PIV, PLIF, shadowgraphy, and Mie Scattering.

Figure 15 – Engine Combustion Network optically accessible combustion vessel



Source: ECN (2018).

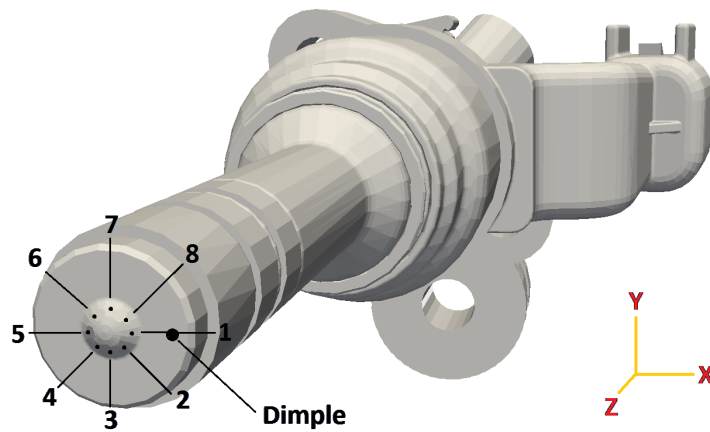
For the spray simulations performed in this work, validation data were obtained from the Engine Combustion Network (ECN), which is an international collaboration among experimental and computational researchers in engine combustion, aimed at establishing well-documented experimental data to be used as a validation source for computational models, as well as providing a framework for collaborative comparison of measured and modelled results and identifying priorities for further

research in the field (ECN, 2018). From the ECN database, two test-cases were selected. The first, named “Spray G”, is used to validate the gasoline non-reactive injection models, whereas the second, named “Spray A”, is employed to validate the combustion models used in this work.

#### 4.2.1 ECN “Spray G”

The ECN “Spray G” condition corresponds to a non-reacting early injection case for spray-guided gasoline injectors. The 8-hole injector, illustrated in Figure 16, is specified for modern injection systems with high pressure capabilities. Experimental validation data concerning the liquid/gaseous penetration length and droplet size/velocity distributions are obtained from Parrish (2014a), whereas measurements of the gaseous phase are available from Sphicas et al. (2017b). Further information regarding the operating conditions of this case is presented in Table 6:

Figure 16 – Spray G injector



Source: ECN (2018).

Table 6 – “Spray G” condition

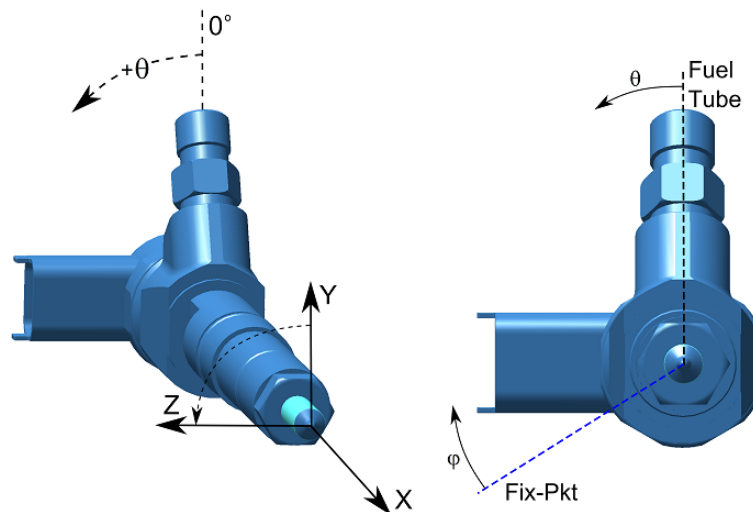
Fuel	Iso-octane
Injection pressure	20 MPa
Fuel temperature	90 °C (363 K)
Ambient temperature	300 °C (573 K)
Ambient density	3.5 kg/m <sup>3</sup>
Ambient pressure	6 bar (600 kPa)
Ambient composition	89.71% N <sub>2</sub> , 6.52% CO <sub>2</sub> , 3.77% H <sub>2</sub> O
Injected quantity	10 mg
Injection duration	780 μs
Number of nozzle holes	8 (equally spaced)
Hole diameter	165 μm
Fully included angle	80°

Source: ECN (2018).

#### 4.2.2 ECN “Spray A”

The ECN “Spray A” condition corresponds to a low-temperature reacting injection case relevant to engines that use moderate EGR (exhaust gases recirculation). Although the single hole injector depicted in Figure 17 is intended to be used for diesel like sprays, using n-dodecane fuel, a few experimental works providing validation data are available in the literature using primary reference fuels (PRF), with varying octane number (PASTOR et al., 2016; JAIN; AGGARWAL, 2018). Further information regarding the exact operating conditions of this case, corresponding to a modified "Spray A" condition, is presented in Table 7:

Figure 17 – Spray A injector



Source: ECN (2018).

Table 7 – Modified “Spray A” condition

Fuel	n-heptane/iso-octane blend
Injection pressure	150 MPa
Fuel temperature	90 °C (363 K)
Ambient temperature	627 °C (900 K)
Ambient density	22.8 kg/m <sup>3</sup>
Ambient pressure	60 bar (6 MPa)
Ambient composition	21% O <sub>2</sub> , 69.33% N <sub>2</sub> , 6.11% CO <sub>2</sub> , 3.56% H <sub>2</sub> O
Injected quantity	6.92 mg
Injection duration	3000 μs
Number of nozzle holes	1 (axially oriented)
Hole diameter	90 μm

Source: ECN (2018).

## 5 RESULTS AND DISCUSSION

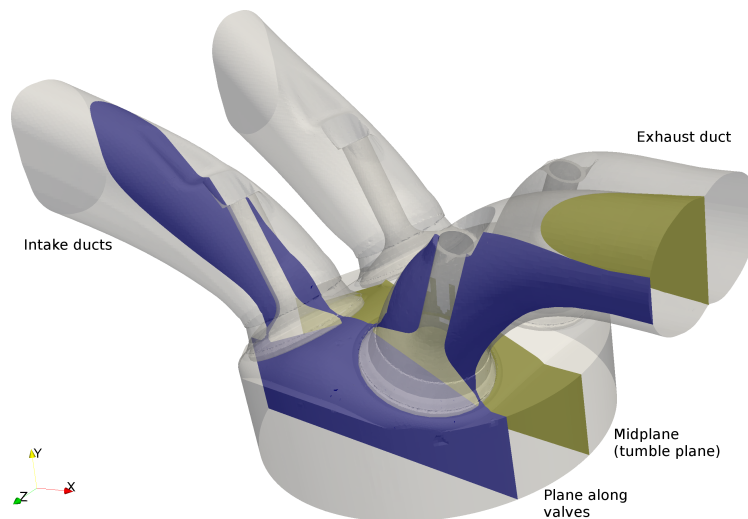
### 5.1 ENGINE FLOW STUDY

In the first section of this chapter, the results regarding the LES of the Darmstadt engine internal flow are presented and compared against experimental values. In the simulations, results are obtained from the average of 20 engine cycles, while the experimental results are an average of 80 different cycles. The solution of one cold-flow engine cycle using 3 computational nodes (72 processors) of the Comet super-computer (SDSC, 2019) took approximately 5 days of computation. Information on the numerical setup can be found in Appendix B.1, and some setup files are included in the Annex.

#### 5.1.1 Flow Visualization

The first results to be presented are planar plots of the velocity components in the radial ( $u$ ) and axial ( $v$ ) directions, in order to analyze the results qualitatively. This analysis is done sampling data from two different planes, one cutting the domain in half and the other showing the flow along two of the valves, as shown in Figure 18. For the midplane, a comparison to measured PIV data obtained by Freudenhammer et al. (2015) is done for the purpose of validation, selecting data from four different instants along the intake and compression strokes. For the plane along the valves, no validation data is available, but the results are shown anyway to qualitatively demonstrate the generation of turbulence around the valves during two instants during the intake stroke and two instants during exhaust.

Figure 18 – Sampling planes



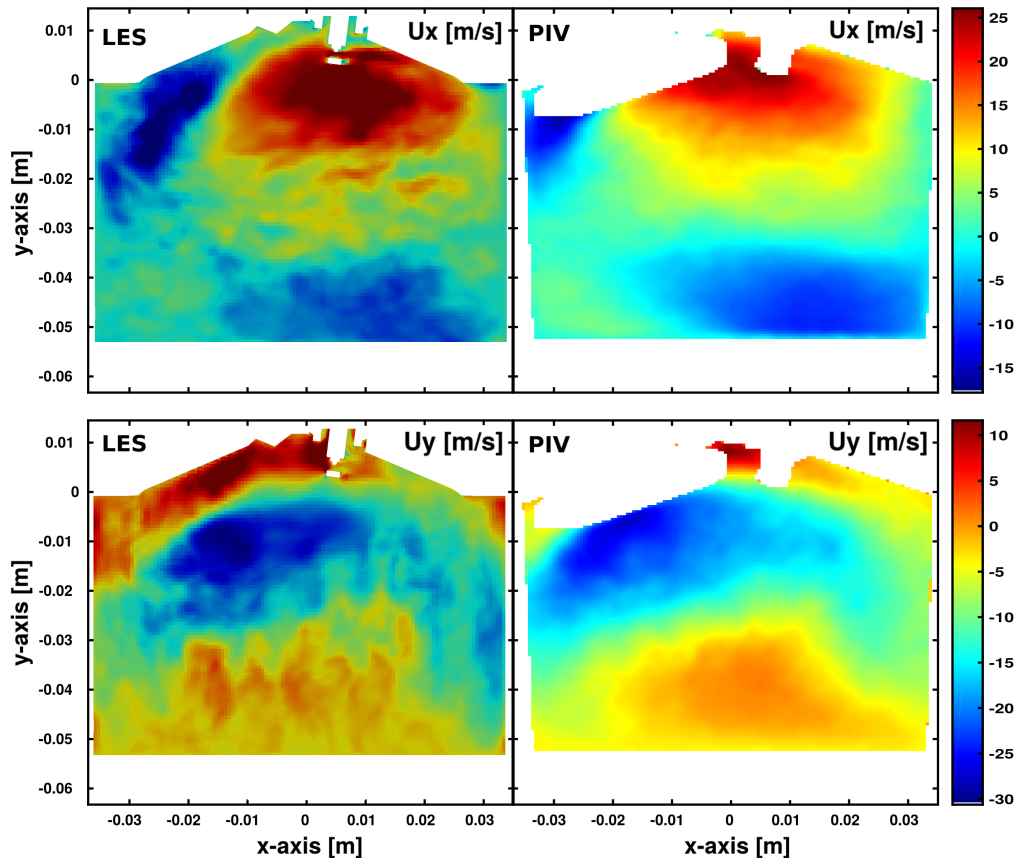
Source: Elaborated by the author.

##### 5.1.1.1 Flow along midplane

The first data to be presented and compared against experimental measurements are taken from the midplane at 270 crank angle degrees before top dead center (270 CAD bTDC), when the piston completes half of the intake stroke, as shown in Figure 19. In this figure, and in the others to follow,

the first row corresponds to radial velocity ( $u$ ) and the second row corresponds to the axial velocity ( $v$ ), whereas the first column presents the results regarding the LES performed in this work and the second column refers to the PIV results from Freudenhammer et al. (2015). Moreover, in order to perform a fair comparison between computational and experimental works, the same color scheme as well as the same value range is used. As during this instant the intake valves are quite low in their lift, one substantial part of the plane cannot be seen in the PIV since the valves cross the line of sight of the camera used to capture the scattered light.

Figure 19 – Velocity field along midplane at 270 CAD bTDC



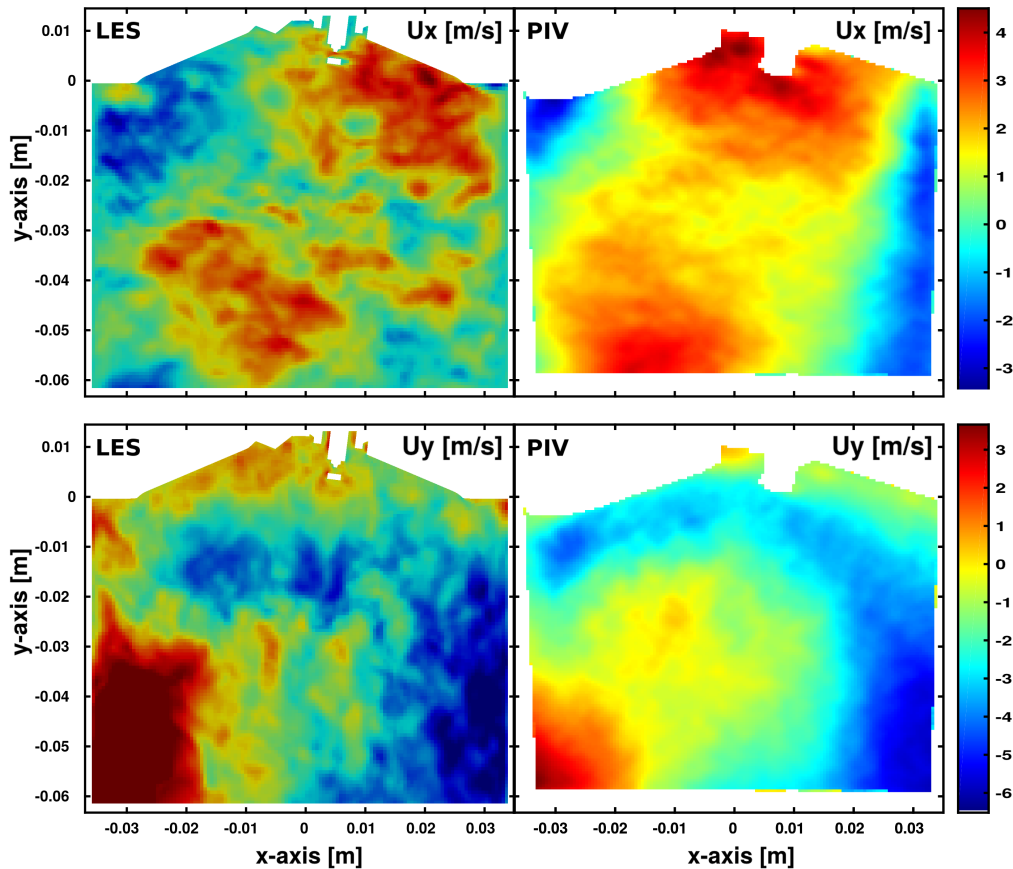
Source: LES (Elaborated by the author). PIV (FREUDENHAMMER et al., 2015).

As it can be seen, although the number of LES cycles is still substantially lower than that obtained experimentally, the LES simulation is capable of capturing the experimental trends reasonably well. The trend one can already observe at this point of the cycle is the expected beginning of the tumble motion in the clockwise direction, since the intake valves are located on the left hand side and the exhaust valves are on the right hand side. Therefore, high radial velocities are present at the top and low radial velocities are seen at the bottom of the plane. At the same time, the axial velocity plots show air flowing from the intake valves to the piston surface, whereas the air directly over the piston is pushed towards the cylinder head.

Moreover, the results for the end of the intake stroke at 180 CAD bTDC are shown in Figure 20, for which instant the intake valves virtually left the line of sight of the camera, thus capturing a measured area similar in size to the plane taken from the computational model. Again the qualitative agreement between simulation and experiment is quite good, showing the capability of the LES simulation in

capturing fluctuations in the flow. In contrast to the results at the previous time, at BDC velocity magnitudes are substantially lower, because the piston stops moving downward, thus reducing the pressure drop effect that sucks the air into the cylinder.

Figure 20 – Velocity field along midplane at 180 CAD bTDC



Source: LES (Elaborated by the author). PIV (FREUDENHAMMER et al., 2015).

After the closure of the valves, at 125 CAD bTDC, the compression stroke starts and results during this event are shown for the instants of 90 CAD bTDC, in Figure 21, and at 45 CAD bTDC, in Figure 22. One more time, the LES results agree very well with the experimental ones. During 90 CAD bTDC, the tumble motion is still very strong, although the valves have been closed for more than 30 CAD. At 45 CAD bTDC, this vortex motion is not as strong as at the previous time as piston approaches TDC, but it is still very noticeable.

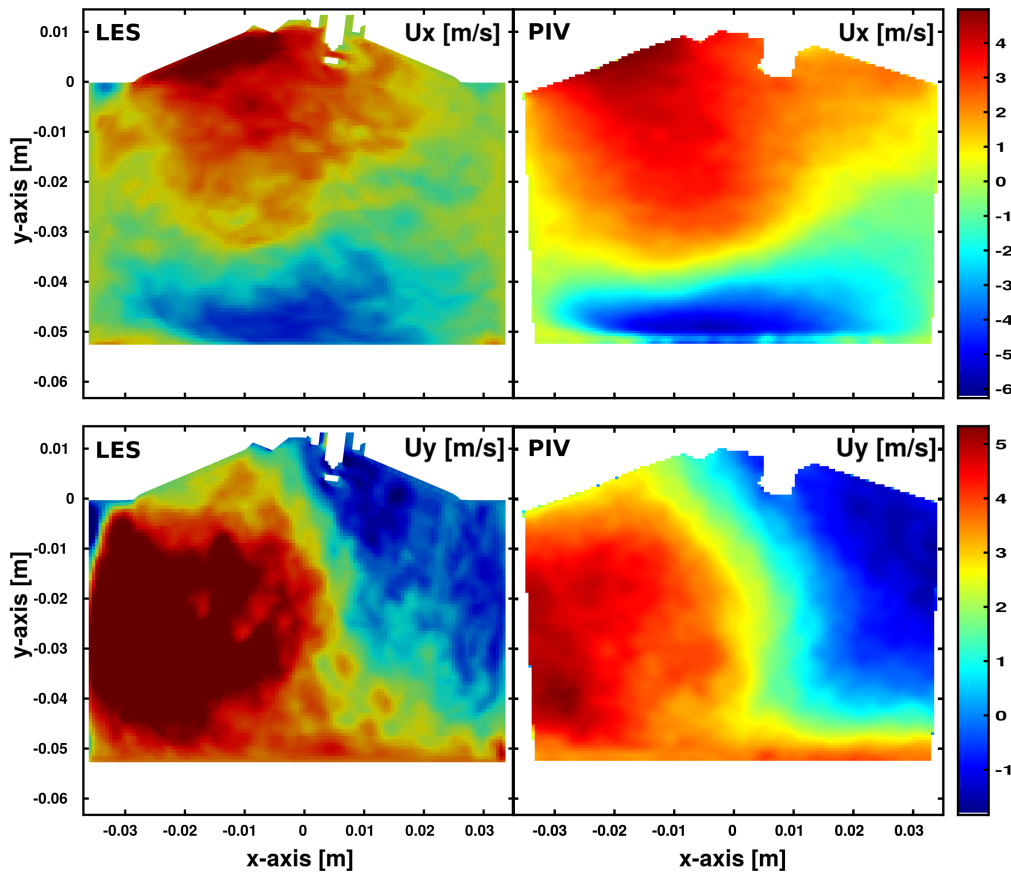
#### 5.1.1.2 Flow along valves

In addition to the results shown above for the midplane, a few plots are presented on the plane crossing the valves. Although there is no validation result for this particular plane, it is interesting to examine what is happening there since the valves are one of the key factors for generating the turbulence seen inside the cylinder.

In Figure 23 the results for the velocity magnitude, temperature, and pressure are shown for the time of 270 CAD bTDC. Streamline trajectories are shown in the velocity magnitude plot in order to reveal the direction of the velocity vectors trajectories. The pressure drop caused by the downward movement of the piston forces a considerable amount of air to flow along the intake duct into the



Figure 21 – Velocity field along midplane at 90 CAD bTDC



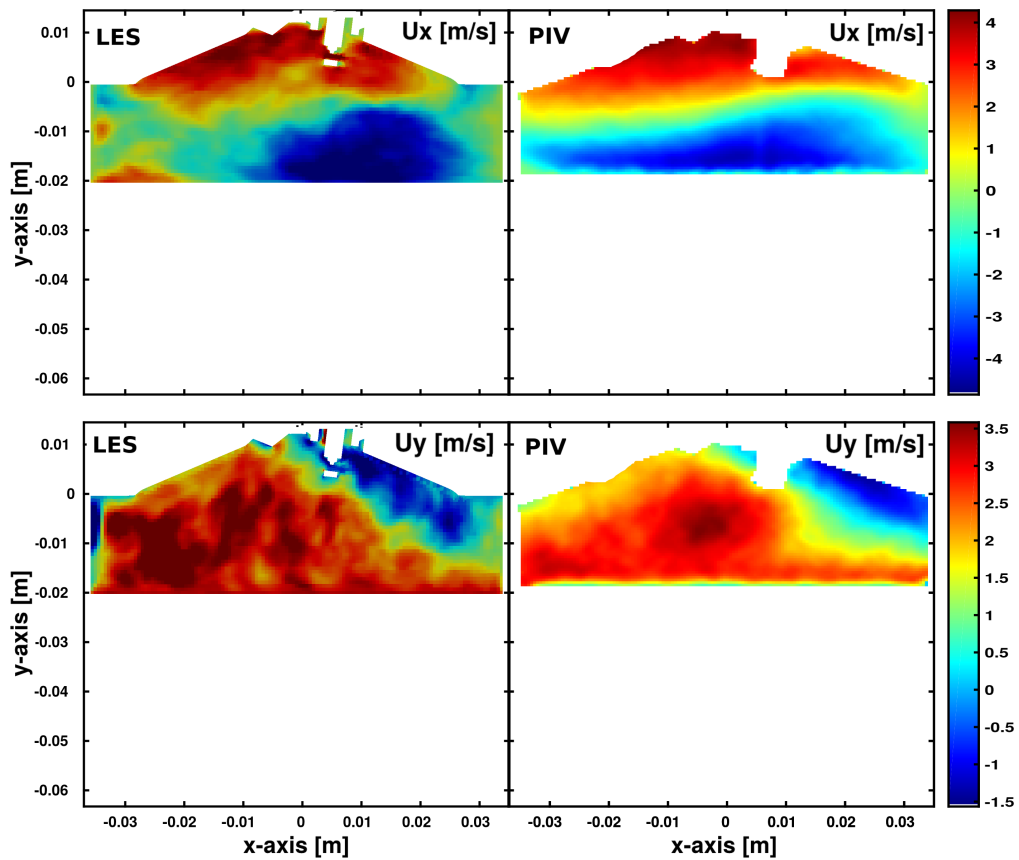
Source: LES (Elaborated by the author). PIV (FREUDENHAMMER et al., 2015).

cylinder. At the point where the intake duct curves to connect with the cylinder head, it is possible to see the phenomenon of flow separation, as the inflowing air jet moves away from the intake duct walls to encounter the top surface of the intake valve, resulting in some reverse flow in the boundary layer region. The subsequent impinging jet formed by the air squeezed between the valve seat and the valve itself produces a strong shear layer, inducing a few vortices along its way. Moreover, the temperature plot shows the significant temperature difference between the cool inflowing air and the temperature of the air in the exhaust pipe.

In Figure 24 the piston is at its lowest position throughout the cycle at 180 CAD bTDC, the valve is still moving towards the valve seat and a strong tumble motion can be seen as air flows in the clockwise direction along the plane. Similar to what was seen above for the midplane, at this point the velocity magnitude is reduced substantially since the piston stops moving down to begin moving up again. From this point on, the amount of turbulence inside the cylinder starts to decay since the piston becomes the only significant source of turbulence as intake valves close at 125 CAD bTDC. After one complete rotation of the crankshaft, at 180 CAD aTDC, as shown in Figure 25, high velocities are seen again inside the cylinder as air is pushed out as the exhaust valves open during the exhaust stroke. Contrary to what happened during the intake stroke, the pressure inside the cylinder is slightly higher than the pressure in the exhaust ducts, just a few degrees after exhaust valves opening.

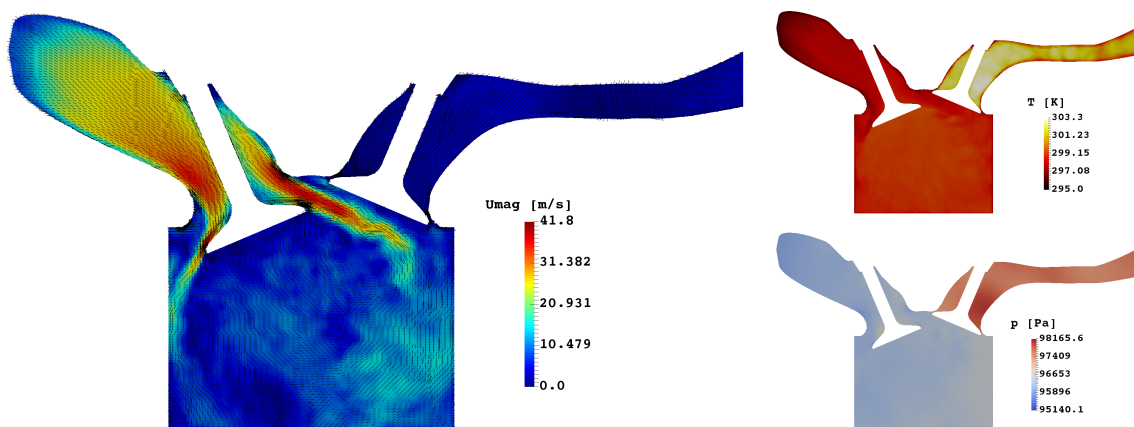
Finally, in Figure 26, the plots for the flow at the end of the exhaust cycle are shown, exhibiting a strong air flow around the exhaust valve towards the exhaust pipe. Temperatures inside the cylinder are

Figure 22 – Velocity field along midplane at 45 CAD bTDC



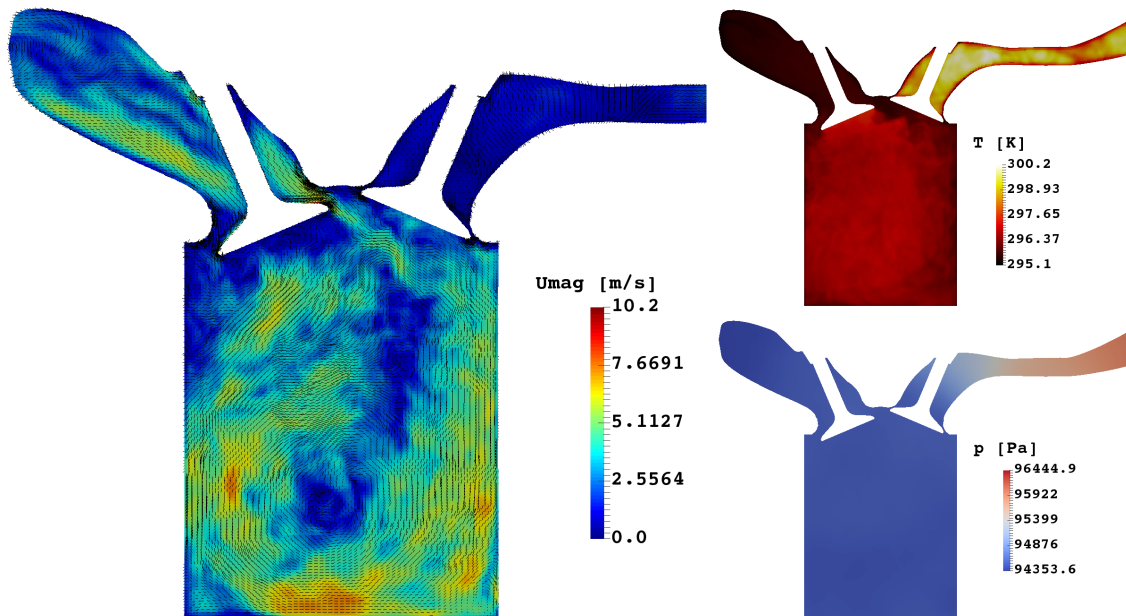
Source: LES (Elaborated by the author). PIV (FREUDENHAMMER et al., 2015).

Figure 23 – Velocity, temperature, and pressure fields along valves at 270 CAD bTDC



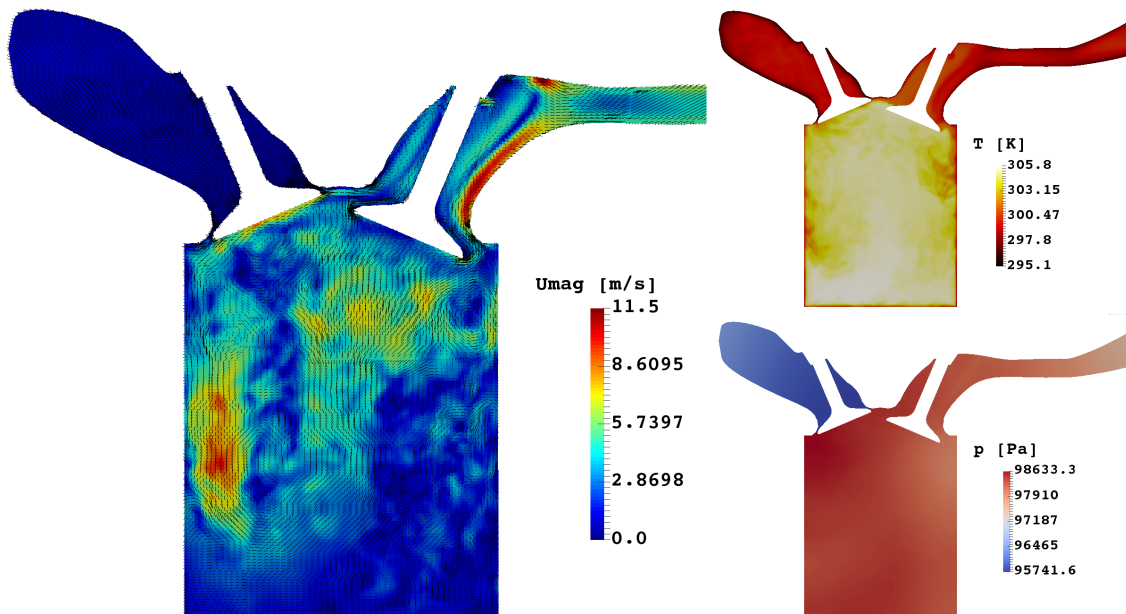
Source: Elaborated by the author.

Figure 24 – Velocity, temperature, and pressure fields along valves at 180 CAD bTDC



Source: Elaborated by the author.

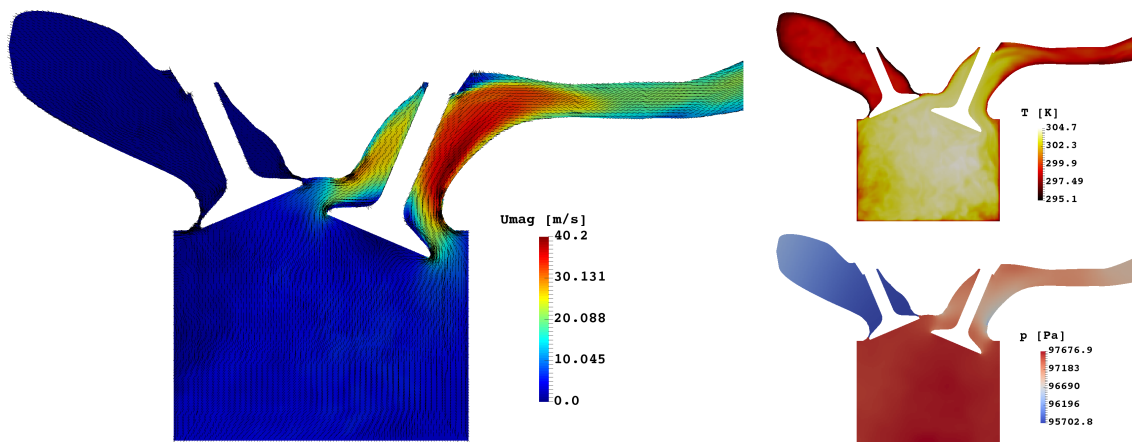
Figure 25 – Velocity, temperature, and pressure fields along valves at 180 CAD aTDC



Source: Elaborated by the author.

still considerably higher than the ones in the intake duct, even several crank angle degrees after the end of the compression stroke.

Figure 26 – Velocity, temperature, and pressure fields along valves at 270 CAD aTDC

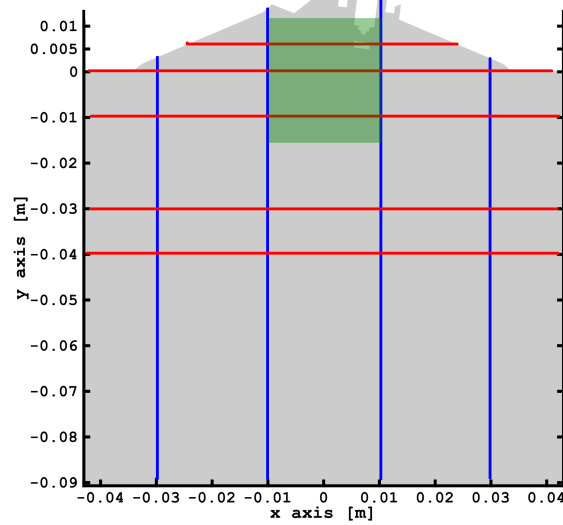


Source: Elaborated by the author.

### 5.1.2 Velocity Field

In this section, a quantitative analysis of the computed velocity field is addressed. For that, some regions are defined for sampling the velocity data from the midplane in four vertical lines (blue), five horizontal lines (red), and one plane (green) in the spark-plug region where probability density functions will be evaluated. All these definitions are shown in Figure 27. Moreover, two quantitative metrics are used to evaluate the level of agreement between simulations and experiments. The first metric is the  $\chi^2$  test, which evaluates the performance of a certain hypothesis (in this case, the numerical model) in representing the expected results (here assumed as the experimental ones) for a certain property. This is done by dividing the square of the deviation  $d$  ( $d = s - e$ , with “s” representing the simulated results and “e” the experimental ones along a particular sampling line) by the expected (experimental) values ( $\chi^2 = d^2/e$ ). The calculated value is then compared to the tabulated value from the  $\chi^2$  distribution for the given degrees of freedom and level of significance  $\alpha$  (here assumed as 5% for a 95% confidence level). If the calculated  $\chi^2$  is smaller than the tabulated value, the hypothesis passes the test; otherwise the hypothesis is rejected. The tabulated maximum value allowed for the model to pass the test is shown in all figures inside a red box. Since this metric is normally applied to cell frequencies (counts) rather than continuous data (like velocity component data) and, consequently, only positive values can be allowed, the sampling points were carefully chosen to not include data ranges where sign changes are present. In order to circumvent this limitation, another metric was also employed to support this analysis, consisting of calculating the error between simulation and experiment using the L2 norm ( $\|u\|_2 = (\sum_{n=i} |u_i|^2)^{1/2}$ ). For a better evaluation of the agreement or disagreement between simulations and experiments, the relative L2 norm is calculated, which can be defined by the ratio of the L2 norm of the deviation ( $d = s - e$ ) to the L2 norm of the experimental results vector. However, another limitation that is present in both metrics is the fact that the yielded error becomes too high if experimental values are close to zero (denominator approaches zero), even when simulated results agree fairly with experimental ones.

Figure 27 – Sampling lines and PDF region



Source: Elaborated by the author.

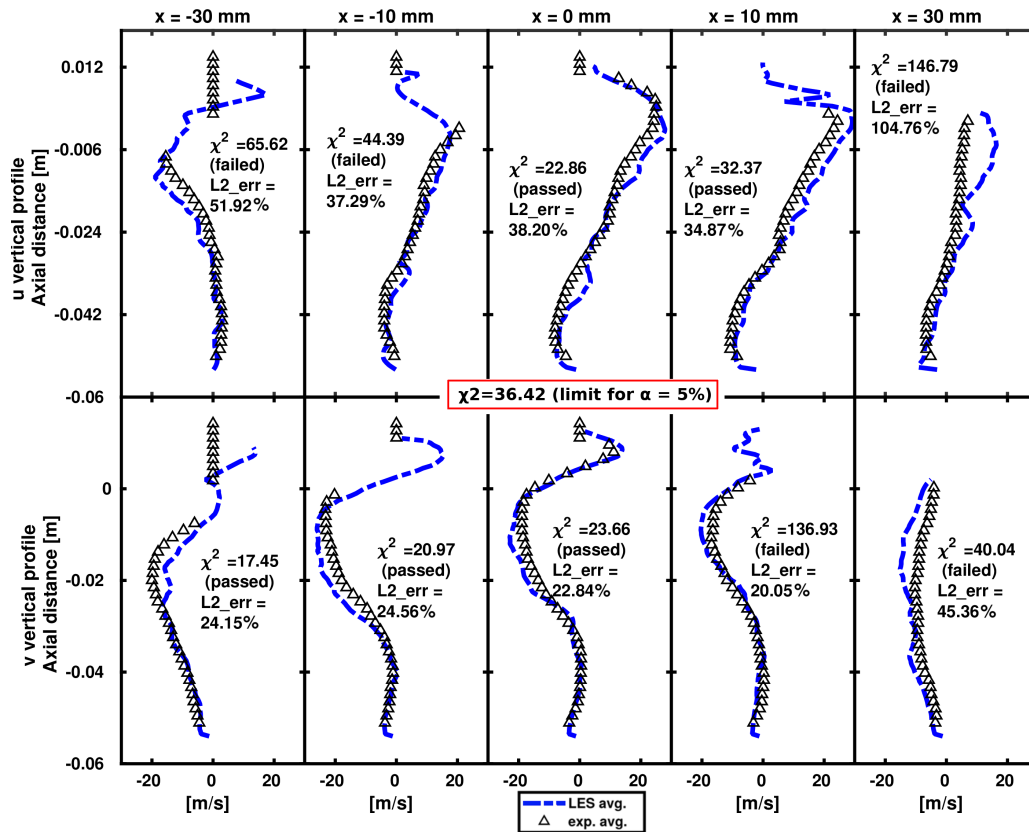
#### 5.1.2.1 Intake Stroke Profiles

Figure 28 presents the vertical profiles at 270 CAD bTDC. The LES results are shown with a dashed blue line, whereas the PIV data from Freudenhammer et al. (2015) are represented with black triangles. The numerical model passed the  $\chi^2$  test with a 95% confidence level in 5 of the 10 profiles shown, with a relative L2 norm error around 30% for the passing profiles. The reasons the model fails the test in some of these profiles can be attributed to inaccuracies in the turbulence model, numerical diffusion and the negligence of important sources of energy loss (such as pumping losses and volumetric inefficiency). Moreover, the standard deviation values of the experimental results are not available for evaluation if the deviations seen in the simulation are within the standard deviation or not, and a comparison of 20 simulated cycles against 80 measured ones can be misleading, since simulation may still converge at another value given more cycles are calculated. Finally, the high calculated  $\chi^2$  values can also be attributed to the inability of the test to deal with continuous data, and some of the profiles where the model failed present a large range of values close to zero, which can negatively affect the performance of the two metrics used.

Figure 29 exhibits similar results at 180 CAD bTDC, with the model passing the test in 6 of the 10 profiles shown. For this particular time, some considerable over prediction of the LES results of the axial velocity at the  $x = 30$  mm sampling line were observed, which can also be noted by the high error values. Since this line is fairly close to the cylinder wall, the disagreement can be explained by inaccuracies produced by the wall model.

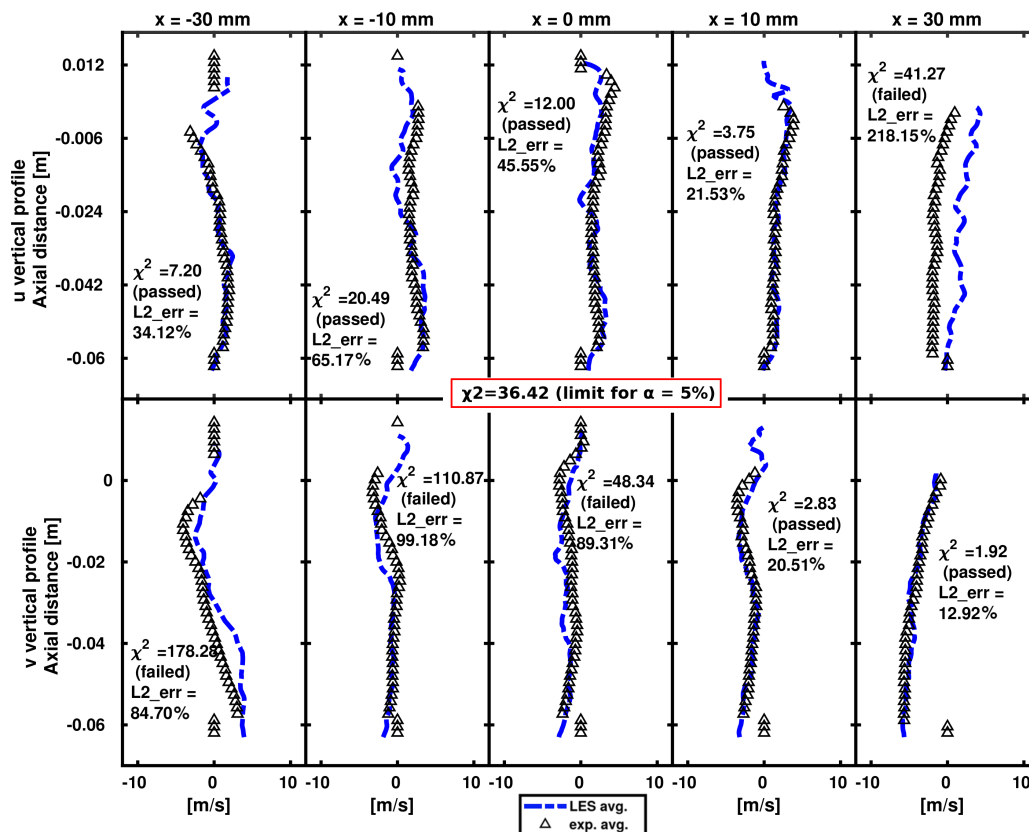
Furthermore, Figure 30 shows the horizontal profiles at 270 CAD bTDC and Figure 31 at 180 CAD bTDC. In these figures, the LES results are shown with a dashed red line. In the horizontal profiles, the model obtains slightly better performance than in the vertical ones, passing the test 6 out of 10 times at 270 CAD bTDC and 8 out of 10 at 180 CAD bTDC. The relative L2 norm error between LES and PIV measurements remains mostly in the range of 20% to 50%, with the high values common in profiles where speeds are around or close to zero.

Figure 28 – Vertical profiles at 270 CAD bTDC



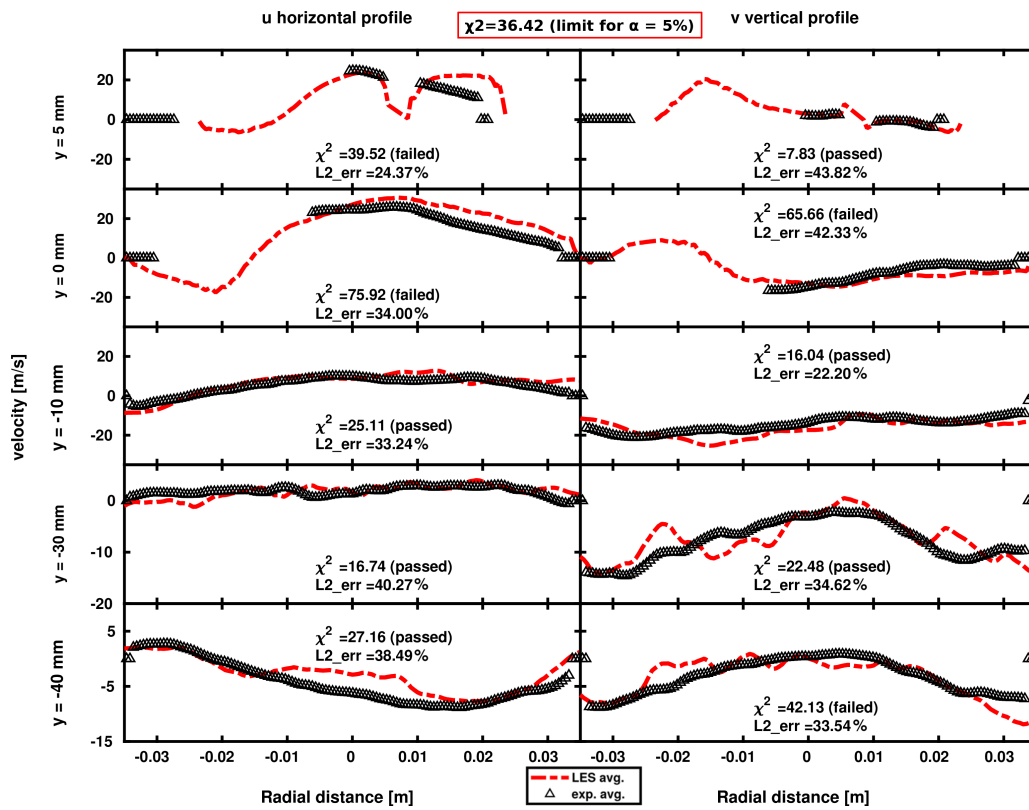
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 29 – Vertical profiles at 180 CAD bTDC



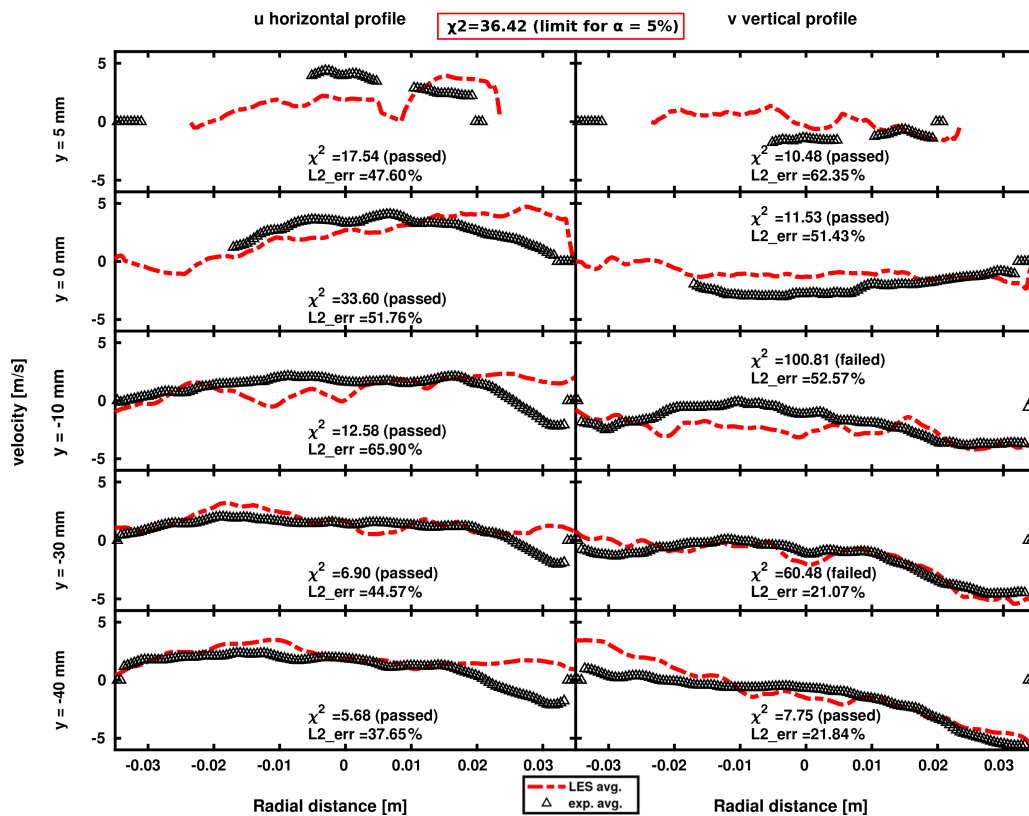
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 30 – Horizontal profiles at 270 CAD bTDC



Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 31 – Horizontal profiles at 180 CAD bTDC

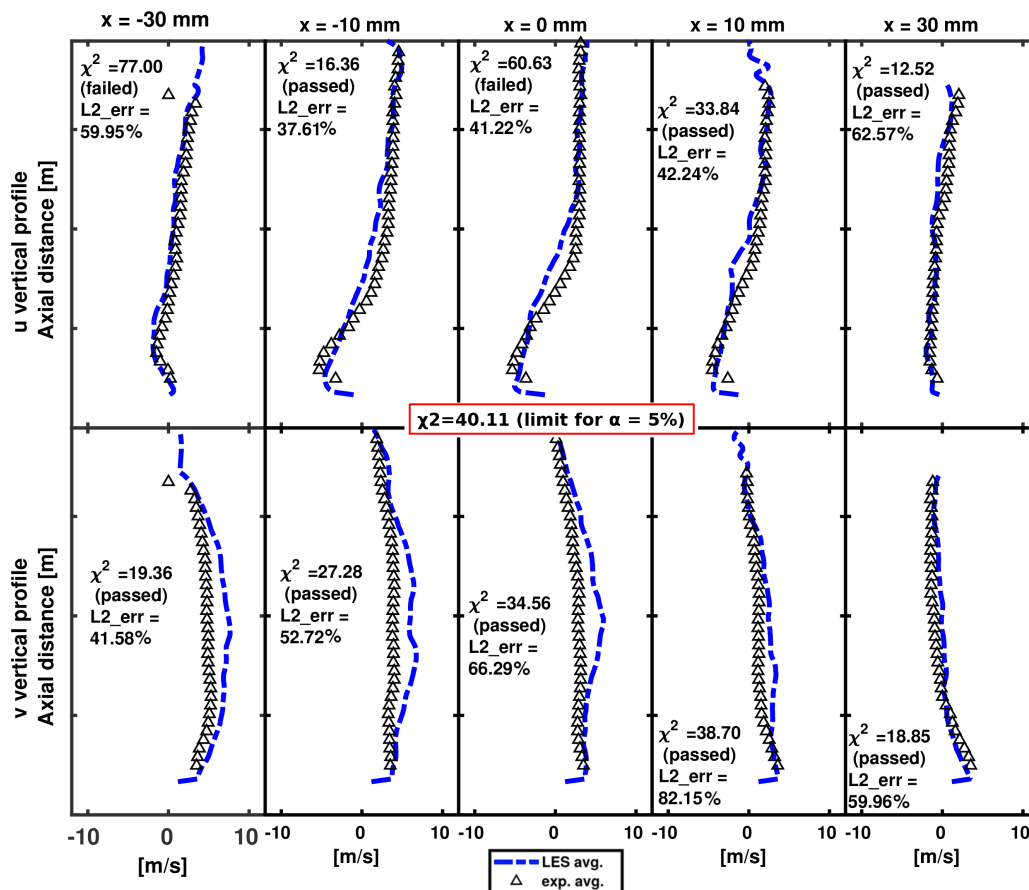


Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

### 5.1.2.2 Compression Stroke Profiles

Following the closure of the intake valves, velocity in both radial and axial directions is considerably lower than during the intake stroke. Vertical samples are shown for 90 CAD bTDC, in Figure 32, and at 45 CAD bTDC, in Figure 33, whereas horizontal profiles for the same times are shown in Figure 34 and Figure 35. Overall, the performance of the numerical model in regard to the  $\chi^2$  test is considerably better during the compression stroke, passing the test in 8 out of 10 vertical profiles at 90 CAD bTDC and 9 out of 10 at 45 CAD bTDC. With respect to the horizontal profiles, the model passes the test in 7 of the 10 profiles at 90 CAD bTDC and in all 6 profiles at 45 CAD bTDC. Even though the model passes the  $\chi^2$  test in many of these profiles, the relative L2 norm error is considerably high due to the decrease in gas velocities in the cylinder after closure of the intake valves. When velocities reach very low values, the performance of such a metric is negatively impacted since the denominator reaches values close to zero.

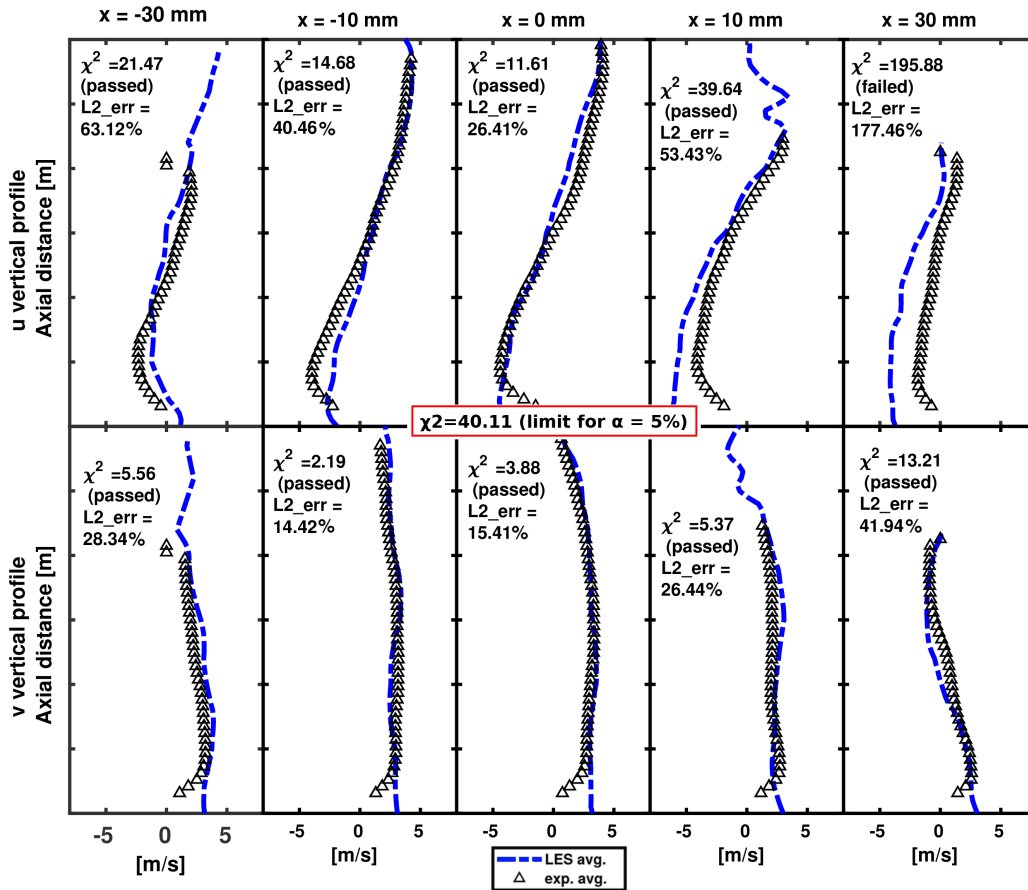
Figure 32 – Vertical profiles at 90 CAD bTDC



Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

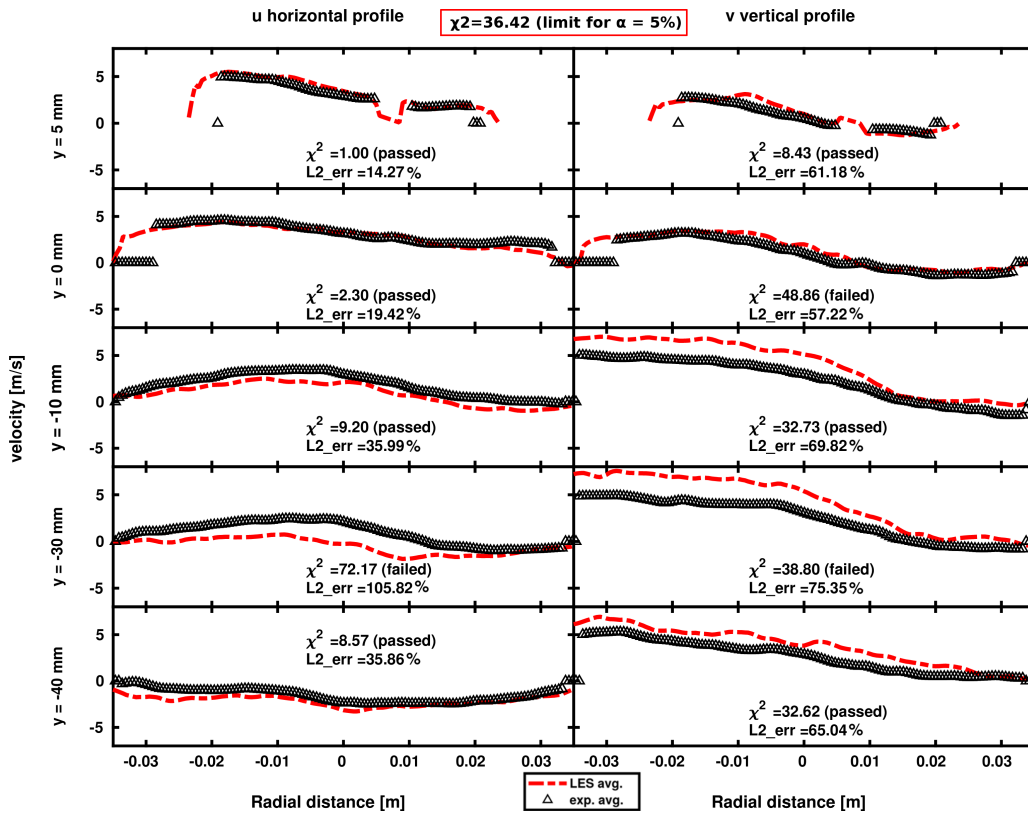


Figure 33 – Vertical profiles at 45 CAD bTDC



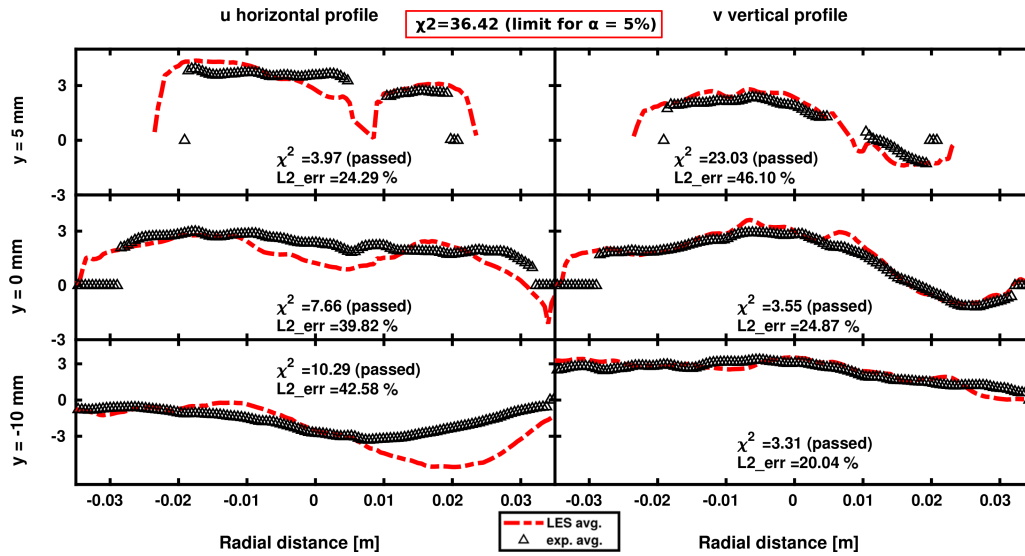
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 34 – Horizontal profiles at 90 CAD bTDC



Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 35 – Horizontal profiles at 45 CAD bTDC

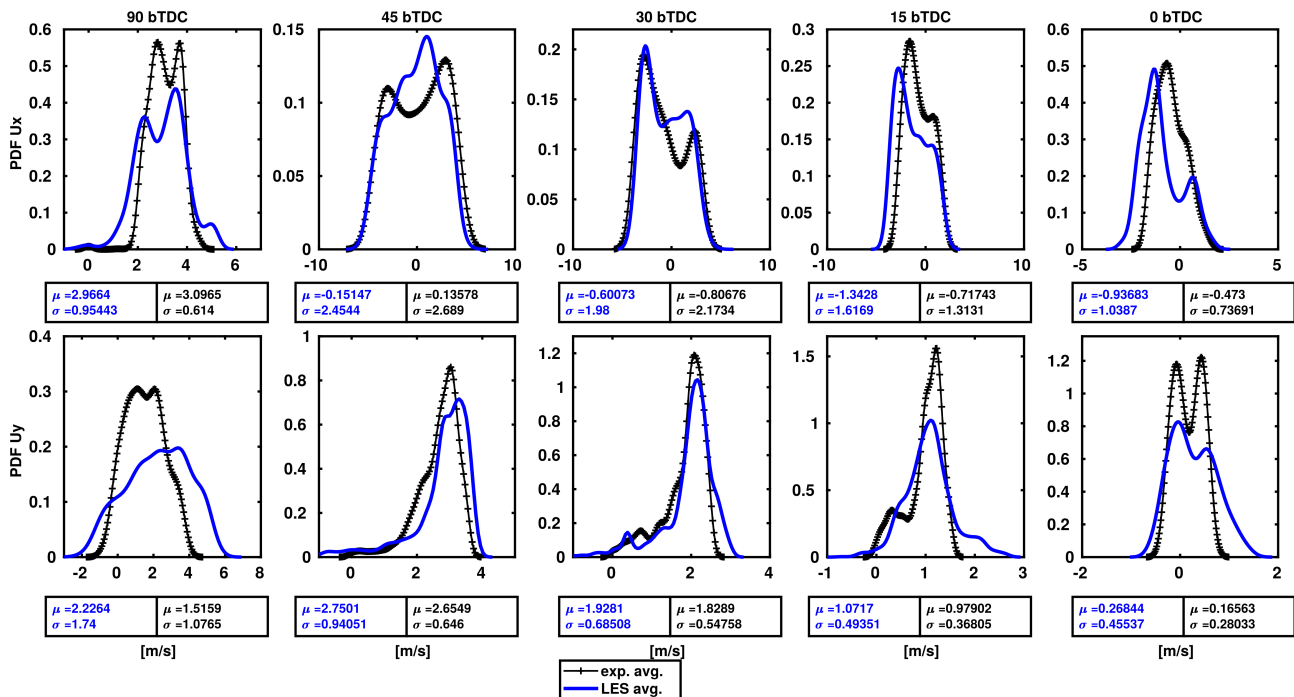


Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

### 5.1.2.3 PDF of the Velocity Field Along Midplane

In order to conclude the quantitative analysis of the velocity field, probability density functions of the entire velocity field in a plane limited by lines  $x = -10$  mm,  $x = 10$  mm and  $y = -15$  mm,  $y = 8$  mm along the midplane are shown in Figure 36. This location is chosen for the particular importance of the turbulent flow around the spark-plug position for mixture formation and initial flame propagation.

Figure 36 – PDF of velocity components during compression stroke



Source: LES (Elaborated by the author). Experiments (FREUDENHAMMER et al., 2015).

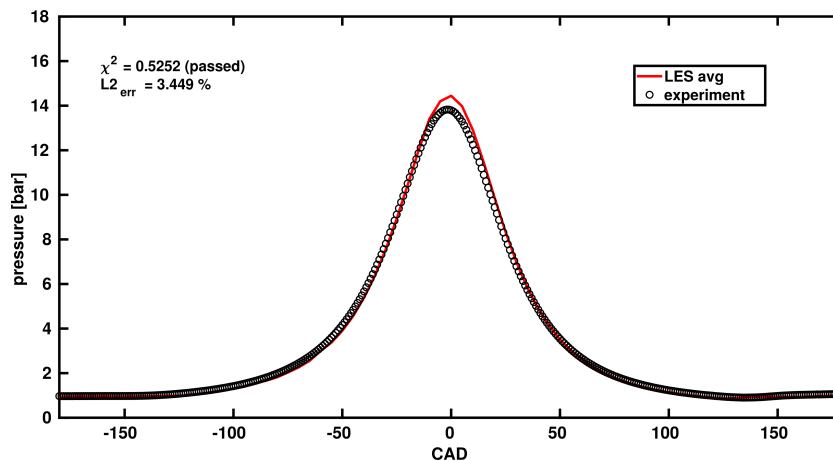
The first row of the aforementioned figure shows results for the radial velocity field, while the second presents the results for the axial component. The PDFs are obtained for instants along the

end of the compression stroke, at 90, 45, 30, 15, and 0 CAD bTDC (bottom dead center). For other flow properties, such as velocity gradients and 2D shear, as well as other time steps, please refer to Appendix C. Overall, PDF shapes from the simulations (blue line) are very similar to the experimental ones (black lines), with mean and standard deviations showing reasonably good agreement for most of the results shown here.

### 5.1.3 Pressure Trace

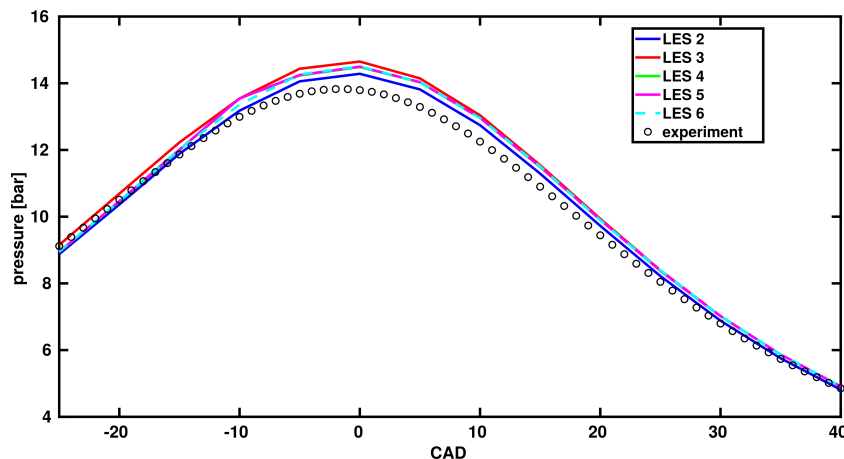
The average in-cylinder pressure trace is shown in Figure 37. The LES results are in quite good accordance with the experimental values (passes  $\chi^2$  test with 95% confidence level and 3.45% relative L2 error), despite the difference in number of cycles (20 against 80). Until 10 CAD bTDC and after 20 CAD aTDC, error values remain below 5%. Within that period, error in respect to the PIV data increases slightly until its maximum value of 6.44% at 5 CAD aTDC. Furthermore, some cyclic variation is seen in the LES, as shown in Figure 38.

Figure 37 – Pressure trace in averaged LES vs. averaged experiment



Source: LES (Elaborated by the author). Experiment (PETERSON et al., 2017b).

Figure 38 – Pressure trace cyclic variations



Source: Elaborated by the author. Experiment (PETERSON et al., 2017b).

### 5.1.4 In-cylinder Turbulence

In order to investigate the phenomenon of turbulence in the combustion chamber, different qualitative and quantitative techniques have been used. First the vortex center tracking algorithm proposed by Graftieaux, Michard and Grosjean (2001) is employed to study the strength of the tumble motion in the spray-guided Darmstadt engine, so that some qualitative clue can be obtained about the development of the vortex along the midplane until its vanishing at TDC. Moreover, the two-point velocity correlation proposed by Janas et al. (2015) will be employed to quantitatively study the phenomenon of tumble breakdown, together with an assessment of the quality of the LES simulation using the criterion proposed by Pope (2011), which quantifies the amount of the turbulent kinetic energy that is resolved.

#### 5.1.4.1 Tumble Motion

The algorithm proposed by Graftieaux, Michard and Grosjean (2001), shown in equation (32), determines the center of the vortex in a bi-dimensional velocity field (for instance, x and y velocity components of Figure 21) by calculating the parameter  $\Gamma$ , for which N is the number of points inside the sampled area S, P is the vortex center, M is a point lying in S, z is the unit vector normal to the plane, and  $U_M$  is the velocity vector. The vortex center lies in the point where  $\Gamma$  is maximum (normalized here as 1).

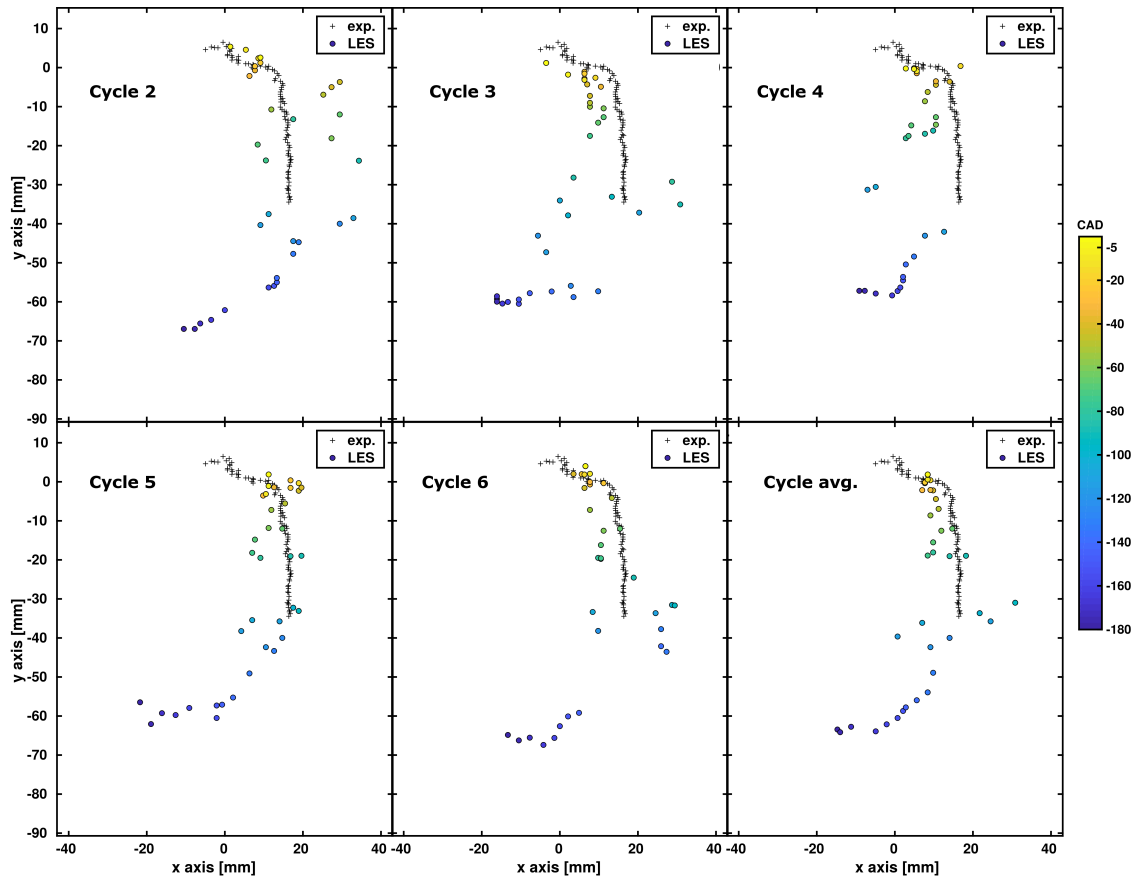
$$\Gamma(p) = \frac{1}{N} \sum_S \frac{(PM \wedge U_M) \cdot z}{\|PM\| \cdot \|U_M\|} \quad (32)$$

The algorithm was applied for different cycles of operation of the LES and also for its averaged velocity field in several surfaces parallel to the tumble plane, ranging from  $z = -20$  mm to  $z = 20$  mm (covering all the space between pairs of intake/exhaust valves). The results for the midplane for five individual cycles and for the average results are shown in Figure 39, together with results obtained from the average velocity field measured by Freudenhammer et al. (2015). In the simulation, the center of the vortex is shown for times ranging between BDC and TDC, while in the experiment the tumble center position is shown from 90 CAD bTDC until TDC.

The path of the tumble core towards TDC for each LES cycle is much more chaotic, with a considerable amount of cyclic variation, than the path taken by the tumble center in the experiment. Even for the average LES cycle, there is a substantial motion in the radial direction, especially after valves closure. The authors believe that these variations are not only due to a lack of LES cycles, but also because the above mentioned equation is rather limited to capture the tumble core if only one plane is taken into account. Turbulence is a three dimensional phenomenon and, therefore, the search for the vortex center solely within the midplane will not be sufficient to find it.

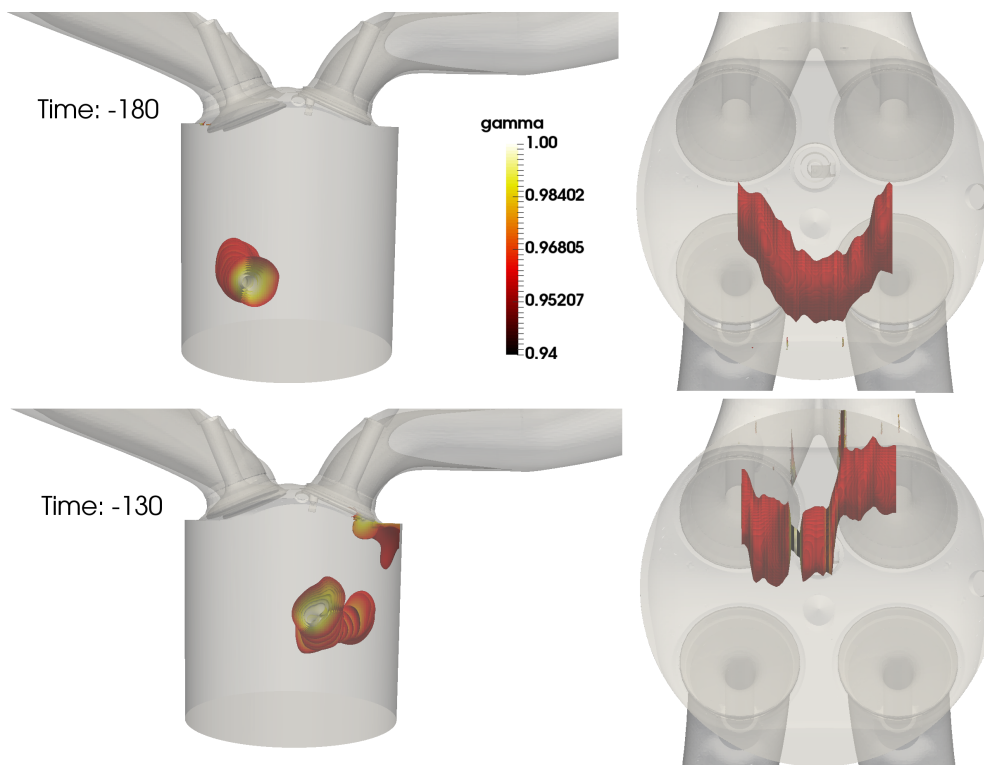
In order to circumvent this issue, the algorithm is applied in several planes along the transverse z axis, as shown in Figure 40 for 180 CAD bTDC and 130 CAD bTDC, using data from the averaged cycles. As it can be noted, this method encounters no problem to predict the tumble core for any of the several surfaces where it was applied at BDC and it stays this way for several crank degrees to come, as suggested in plots of Figure 39. However, the intake valves start to approach the end of their lift and the gap around them starts to get smaller as time advances. This forces the air flowing around the

Figure 39 – Tumble vortex center tracking



Source: LES (Elaborated by the author). Experiment (FREUDENHAMMER et al., 2015).

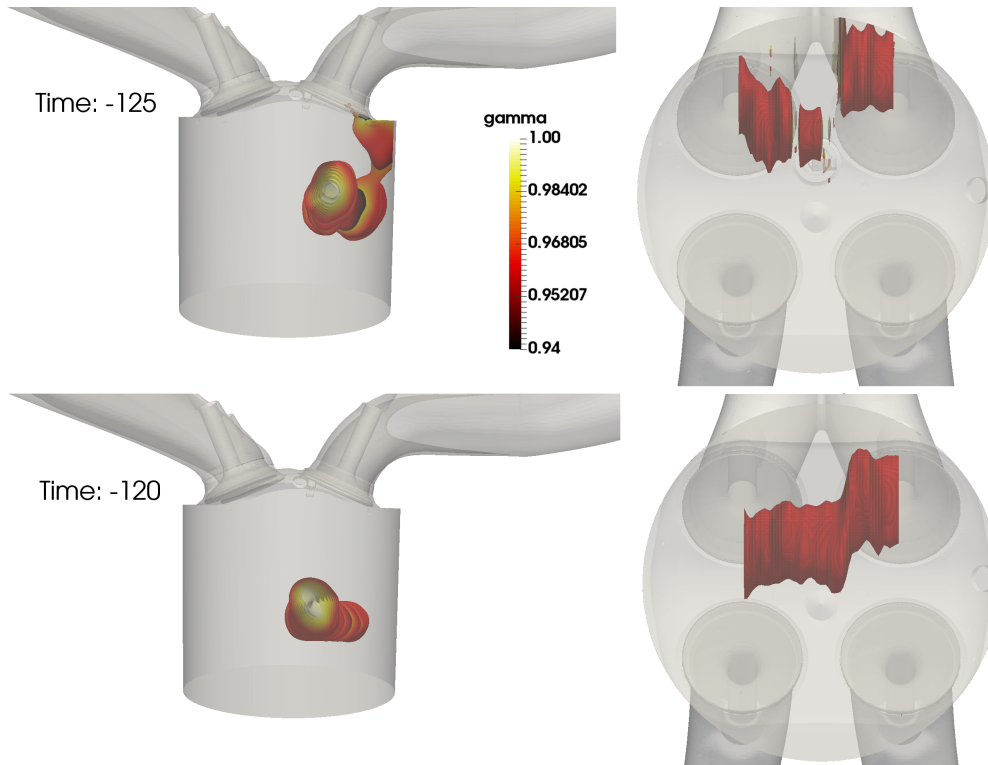
Figure 40 – Tumble core from 180 to 130 CAD bTDC



Source: Elaborated by the author.

valves to gain momentum since the pressure in the combustion chamber is still reasonably lower than inside the intake ducts. The consequence of that is a disruption of the tumble motion in the transverse direction and, therefore, its core cannot be found in some of the planes.

Figure 41 – Tumble core during and after intake valves closure



Source: Elaborated by the author.

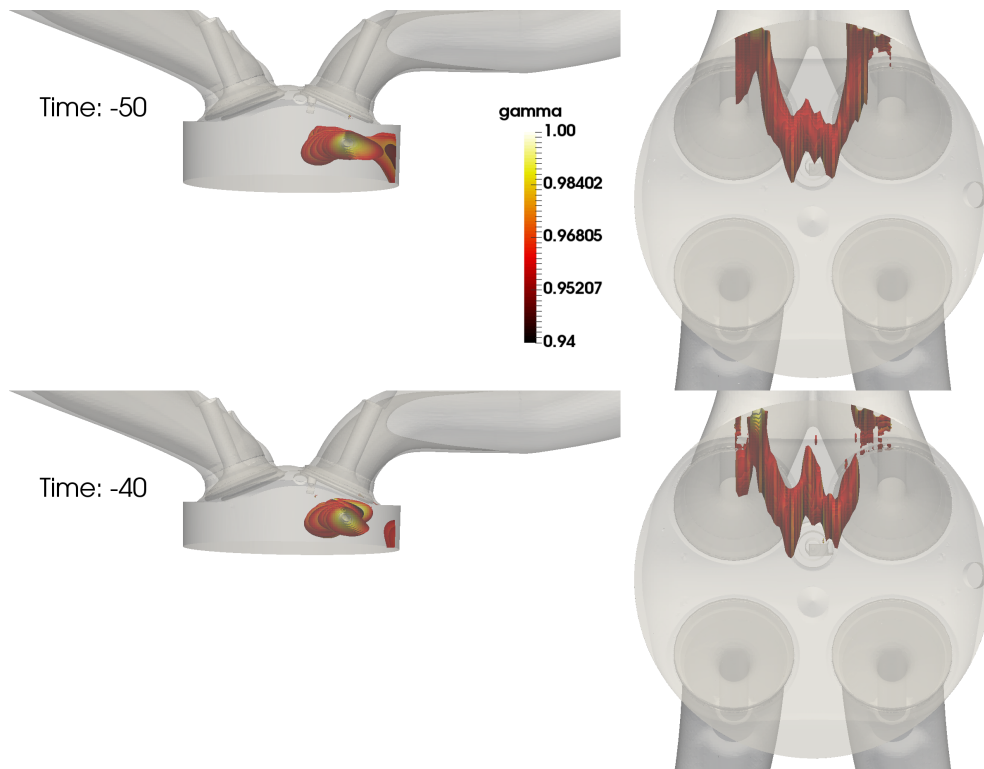
This effect is intensified until the complete closure of the valves at 125 CAD bTDC, as seen in Figure 41. However, only 5 CAD after that, at 120 CAD bTDC, this phenomenon seems to cease as the cylinder region is entirely isolated from the intake/exhaust ducts. Finally, with further advance in time, since the only source of turbulence left is the upward moving piston, the tumble vortex starts to vanish, beginning at 50 CAD bTDC (Figure 42). From this moment on, the tumble core decreases continually in size and the algorithm struggles to find a solution for some of the planes, which can be explained by the low reminiscent levels of in-cylinder turbulence (Figure 43).

#### 5.1.4.2 Two-point Correlation

As a way of supporting the findings using the qualitative approach discussed above, the two-point correlation of Pope (2011), modified by Janas et al. (2015), is employed to determine the intensity of the tumble motion by calculating how correlated velocity data points are to each other along a radial line on the tumble plane. The classic two-point correlation can only be applied if enough statistics for a fully converged result is achieved, which is far from true in this case. However, the modification of Janas et al. (2015) considers the coherent structures instead of the turbulent fluctuations, as in (33):

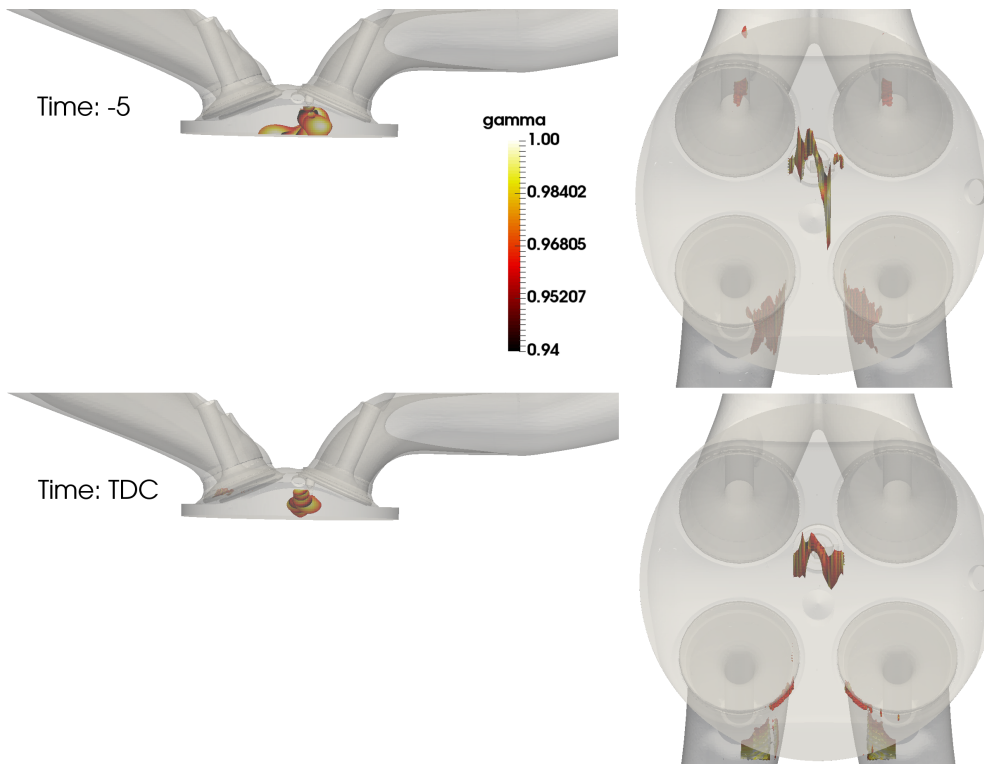
$$R_{\alpha}^{\delta} = \frac{\langle u_{\alpha}^{\delta}(x)u_{\alpha}^{\delta}(x+r) \rangle}{\langle u_{\alpha}^{\delta}(x)^2 \rangle} \quad (33)$$

Figure 42 – Tumble core reduction during compression stroke



Source: Elaborated by the author.

Figure 43 – Tumble core break-down



Source: Elaborated by the author.

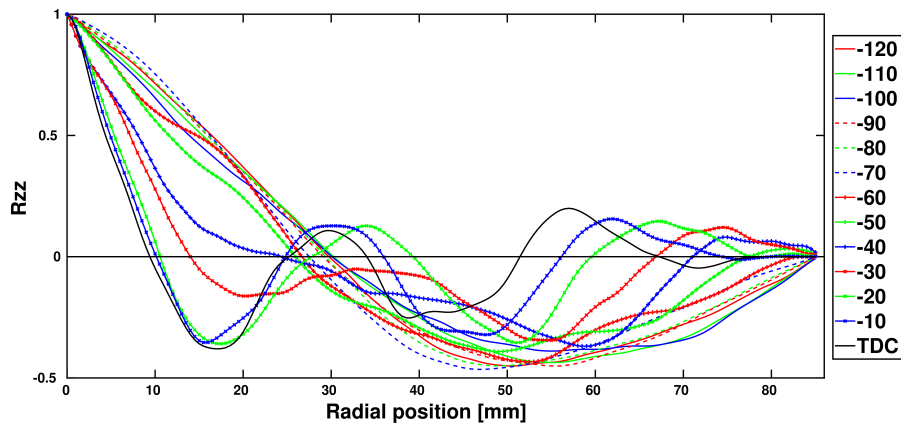
in which the symbol  $u_\alpha^\delta(x)$  stands for the deviation of the resolved velocity  $\tilde{u}$  to the spatial mean  $\bar{u}$  along a particular horizontal line within the tumble plane. A perfect correlation of 1 happens if the velocities in two points are equal, which is true for  $r = 0$ . When this expression is applied at different horizontal lines at different times towards TDC, a fast decay from  $R = 1$  to  $R = 0$  is observed when a weak tumble motion (many small vortices) is present, while for a strong tumble motion this decay should happen more slowly (anisotropic turbulence). From this correlation, it is also possible to derive a length scale for the coherent turbulent structures, using the expression in equation (34):

$$L_\delta = \int_0^q R_\alpha^\delta(r) dr \quad (34)$$

in which  $L_\delta$  represents the length scales of the turbulent structures calculated from the cross-correlation  $R_\alpha^\delta$ , with  $q$  as the distance where the autocorrelation function firstly intersects with the zero line (JANAS et al., 2015).

For the autocorrelation functions, shown in Figure 44, the axial velocity data are sampled along the bore diameter line located at half the distance between piston and cylinder head position at any given time ranging from 120 CAD bTDC to TDC. In Figure 45, the length scales are calculated by integrating the values of the velocity correlation  $R_{zz}$  from  $x = 0$  mm until the point where it first intersects the 0 line. As expected, the closer the piston is from TDC, the faster the autocorrelation values hit zero, which translates to smaller turbulent structures as time passes by (tumble breakdown).

Figure 44 – Two point velocity correlation

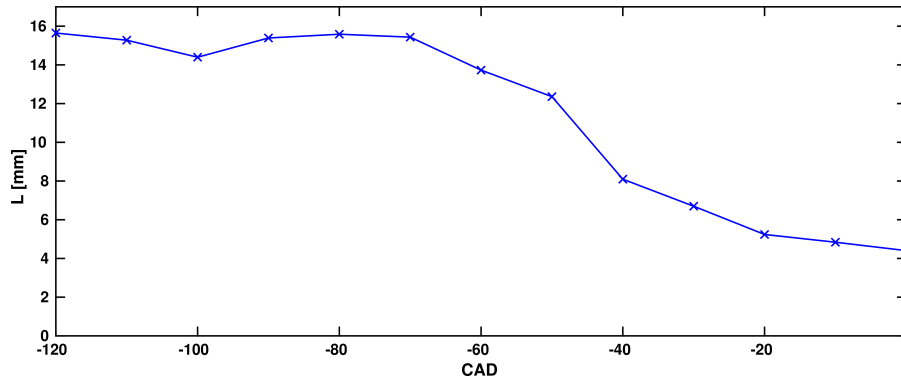


Source: Elaborated by the author.

The characteristic length scale of the turbulent structures remain more or less constant around 15 mm until the time of 70 CAD bTDC. From that point on the length scale reduces in size continuously. The highest rate of reduction happens between 50 CAD bTDC and 40 CAD bTDC, going from something slightly bigger than 12 mm to 8 mm. After that, length scales continue to decrease but at a less pronounced rate, achieving 4.8 mm at 20 CAD bTDC and 4.4 mm at TDC. These results are consistent with the ones shown in the previous subsection, with the length scales starting to decrease at around the same time the tumble core is seen to be affected by the approaching piston after 70 CAD bTDC, as well as with the shrinking of the tumble core as the piston approaches TDC.



Figure 45 – Length scale of coherent turbulent structures



Source: Elaborated by the author.

### 5.1.4.3 LES Quality Criterion

Pope (2011) proposes a criterion to assess the quality of an LES simulation, dividing the amount of turbulent kinetic energy calculated by the sub-grid model in a cell element by the resolved turbulent kinetic energy in that cell, as shown in (35):

$$M(x, t) \equiv \frac{k_r(x, t)}{K(x, t) + k_r(x, t)} \quad (35)$$

in which  $k_r(x, t)$  refers to the sub-grid turbulent kinetic energy, whereas  $K(x, t)$  refers to the resolved kinetic energy at specific location and time. A value of  $M(x, t)$  equal to zero means DNS, while  $M(x, t) = 1$  means RANS. A good quality LES simulation should not exceed 0.2, which means that on average only 20% of the turbulent kinetic energy should be modelled (POPE, 2004).

A snapshot of the  $M$  field during compression is shown in Figure 46, whereas the averaged  $M$  value along the entire four-stroke cycle averaged over 20 cycles is shown in Figure 47. The latter plot shows that the criterion is satisfied almost during the entire cycle, showing undesirably high peaks only during a short period between the intake and compression strokes, whereas the former points out that high  $M$  values can also be obtained locally at specific time steps, which is more evident close to walls, since the need of special models at those regions.

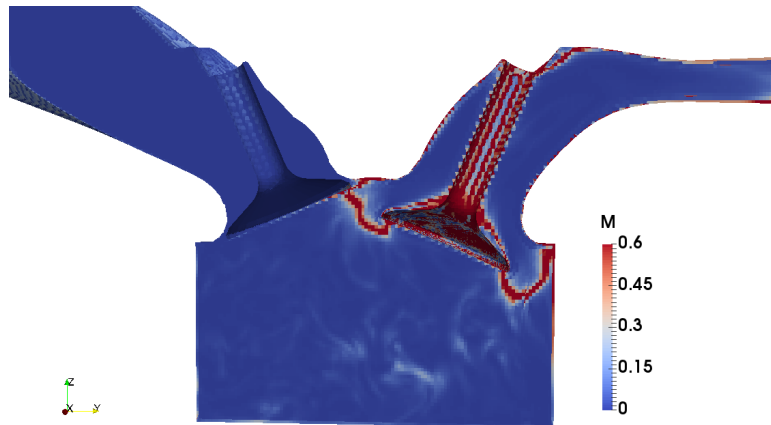
## 5.2 FUEL SPRAY STUDY

In this section, the results concerning the LES of the ECN “Spray G” condition are presented, with the intention to test the Lagrangian-Eulerian packages of the code against experiments. Information on the setup can be found in Appendix B.2, and some setup files are included in the Annex.

### 5.2.1 Flow Visualization

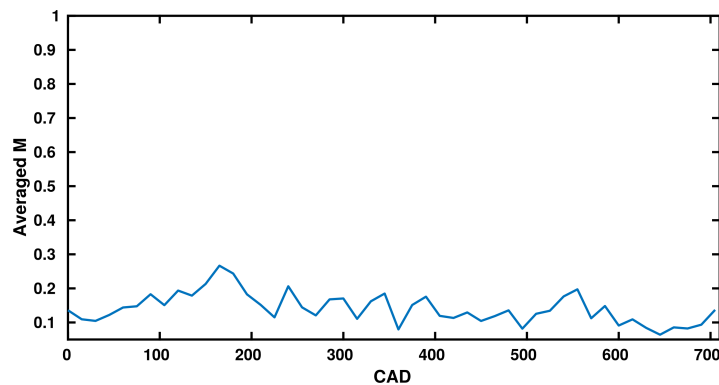
The solution of one cycle of the Spray G non-reactive injection case took approximately 1 day of computation using a local Dell Precision T7910 workstation with 2 Intel Xeon E5-2609 v3 processors (12 computational cores). The spray simulation comprehends a time of 2 ms, from which almost 0.8 ms is the injection duration. In this way, one cannot only see the several phenomena taking place

Figure 46 – M field



Source: Elaborated by the author.

Figure 47 – M plot



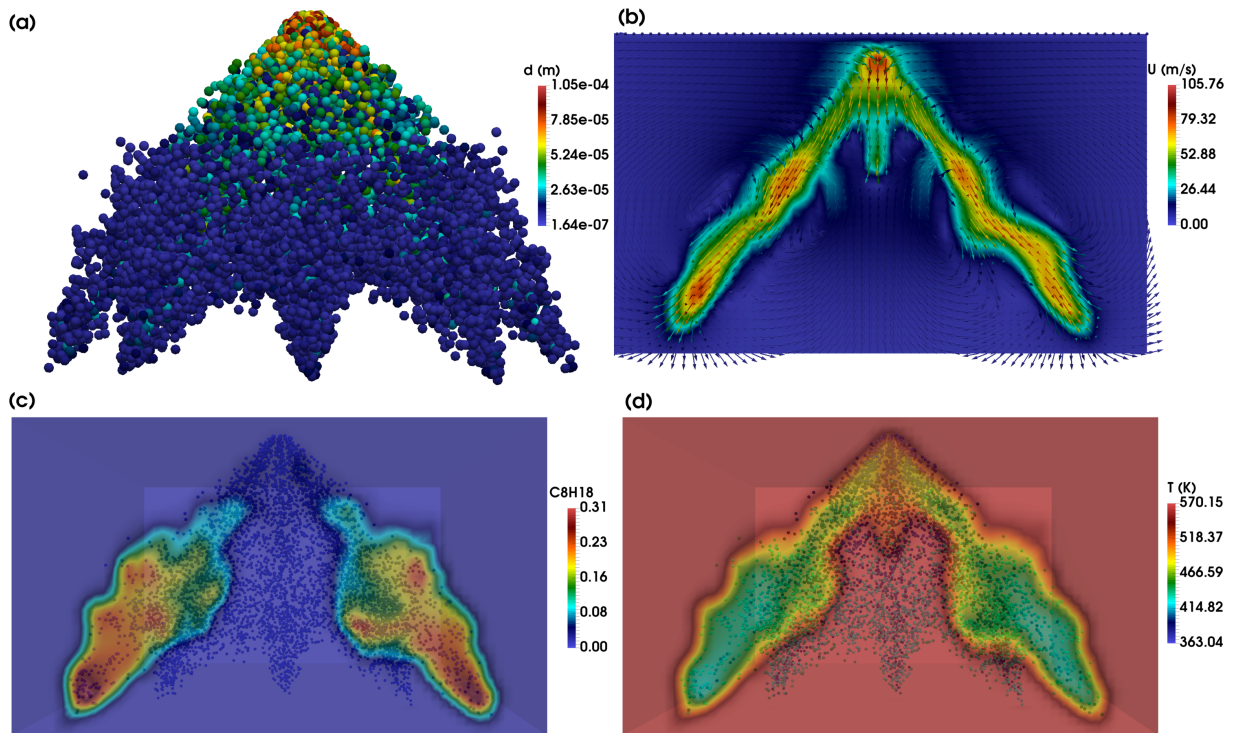
Source: Elaborated by the author.

during injection, but also the mixture formation after the needle is closed. Thus, some plots of the spray simulation in four different instants at times 0.3, 0.6, 1 and 1.4 ms after start of injection (ASOI) are shown in Figure 48 and Figure 49, before end of injection, and Figure 50 and Figure 51, after end of injection.

At the beginning of the injection, a dense spray cloud is seen developing from the injector tip toward the walls of the domain, exchanging momentum and heat with the gaseous phase. Due to evaporation, already at 0.3 ms some considerable amount of fuel can be found in the gaseous phase although all the injected plumes still keep a solid conical shape, as seen in the Lagrangian particle plot of Figure 48. As the injected parcels travel further downstream into the domain, high breakup rates are observed at 0.6 ms due to the strong aero-dynamical forces experienced by the droplets. Therefore, the evaporation process starts to intensify at the tip of the jet, disrupting the spray cloud as droplets move down, as seen in Figure 49.

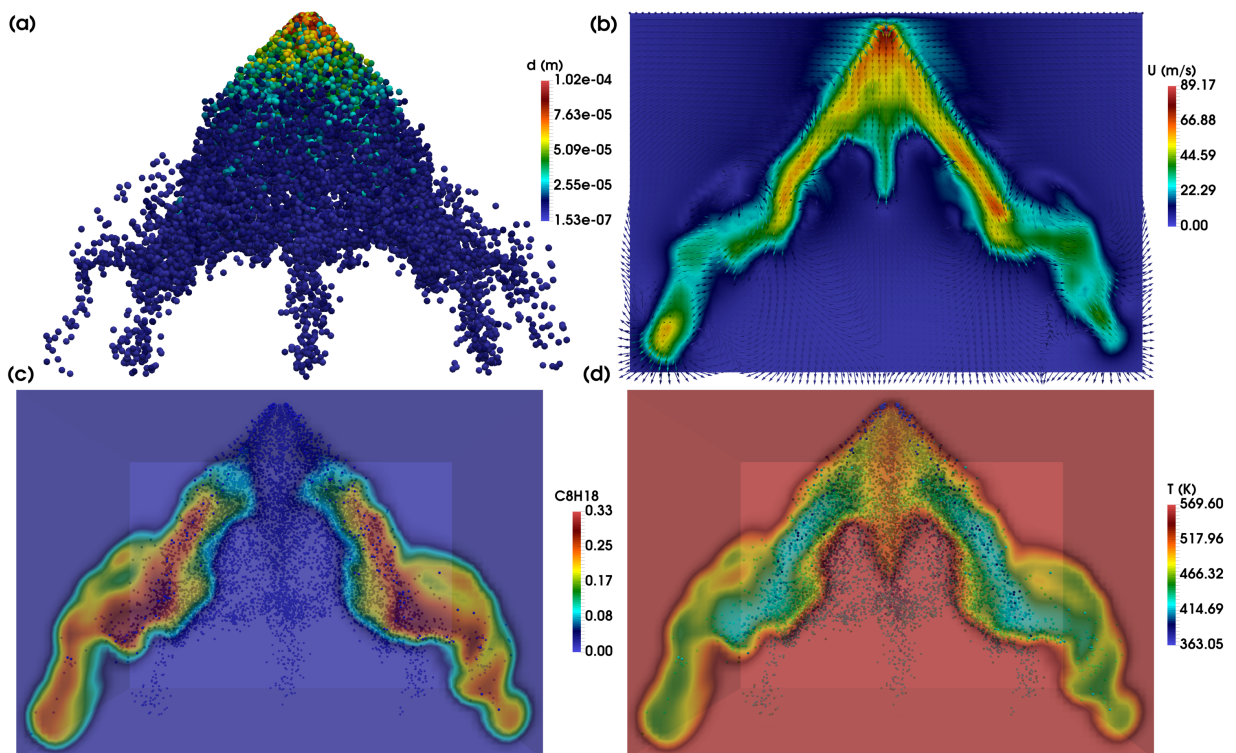
After injection ends at 0.78 ms, the remaining droplets still carry a lot of momentum as they are being slowed down by the gas at 1.0 ms. Moreover, gaseous temperature reduces even more due to the evaporative cooling effect caused by the spray, as shown in Figure 50. Finally, at 1.4 ms only a few droplets are found in the domain, as the moment of complete liquid fuel vanishing gets closer. All droplets at this time are smaller than  $10 \mu\text{m}$ , as seen in Figure 51.

Figure 48 – Spray simulation at 0.3 ms



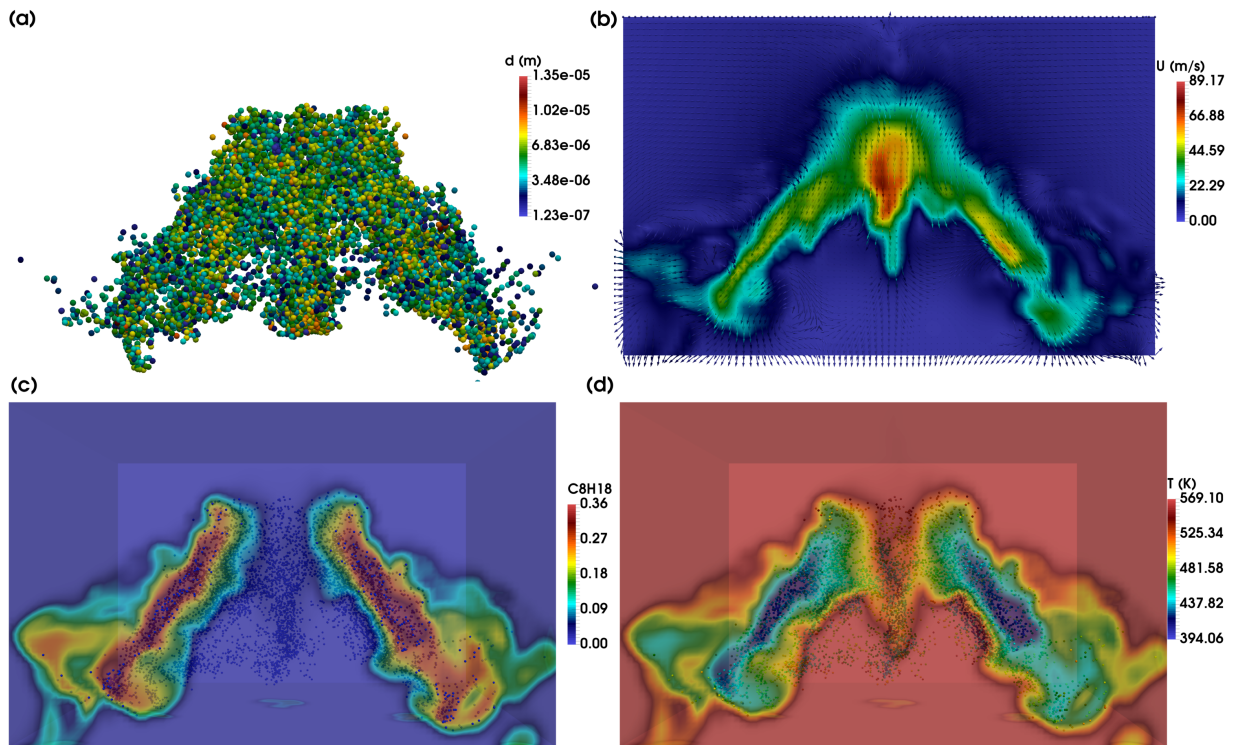
Source: Elaborated by the author.

Figure 49 – Spray simulation at 0.6 ms



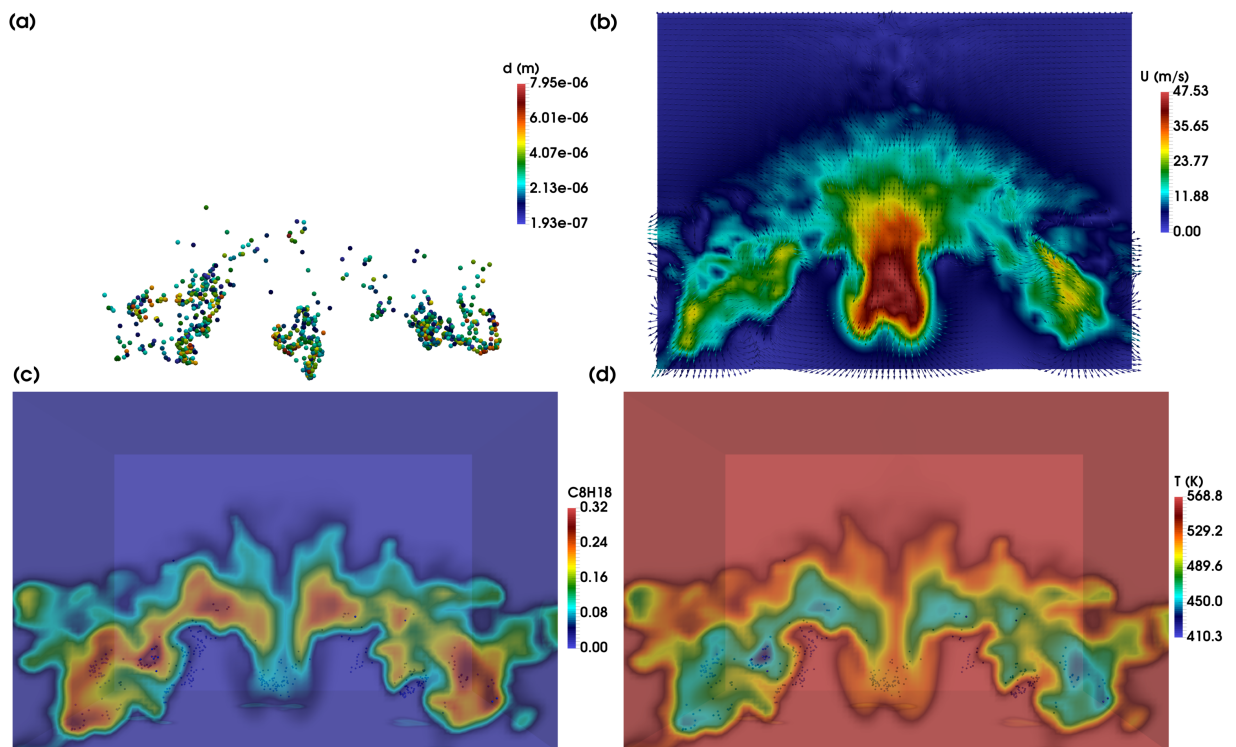
Source: Elaborated by the author.

Figure 50 – Spray simulation at 1.0 ms



Source: Elaborated by the author.

Figure 51 – Spray simulation at 1.4 ms



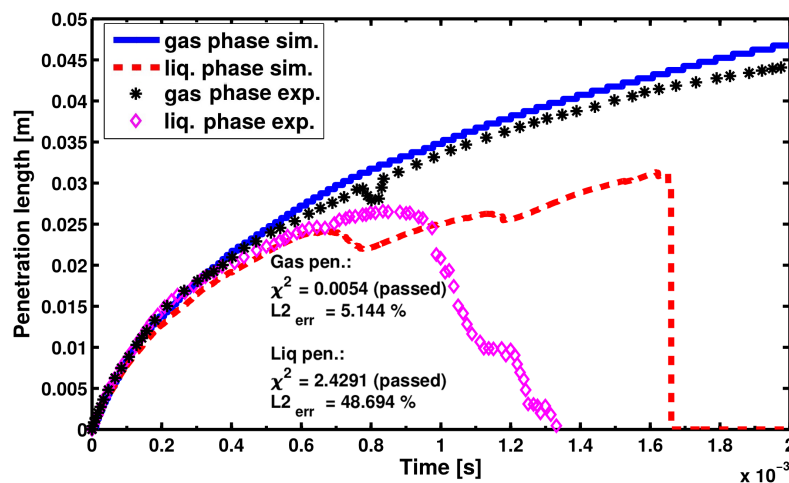
Source: Elaborated by the author.

## 5.2.2 Results along time

### 5.2.2.1 Penetration Length

In Figure 52, the penetration length for both liquid and gas phases is presented and validated against experimental data. The agreement with the experiment is quite good for the gaseous (passes  $\chi^2$  test with 95% confidence level and 5.14% relative L2 error) as well as for the liquid penetration length until end of injection (0.78 ms). After that, the simulated gaseous penetration length starts to slightly over predict the measured data until end of simulation, whereas the liquid length diverges highly from the experiment, taking more than 0.2 ms for all liquid phase to vanish. For the liquid phase, the  $\chi^2$  test was applied only until the time of 1.3 ms, since after that experimental values goes to zero. In this case, the model passes the test, even though there is a clear disagreement after end of injection and a relative L2 error of 48.69%.

Figure 52 – Penetration length for liquid/gas phases validated against experimental data

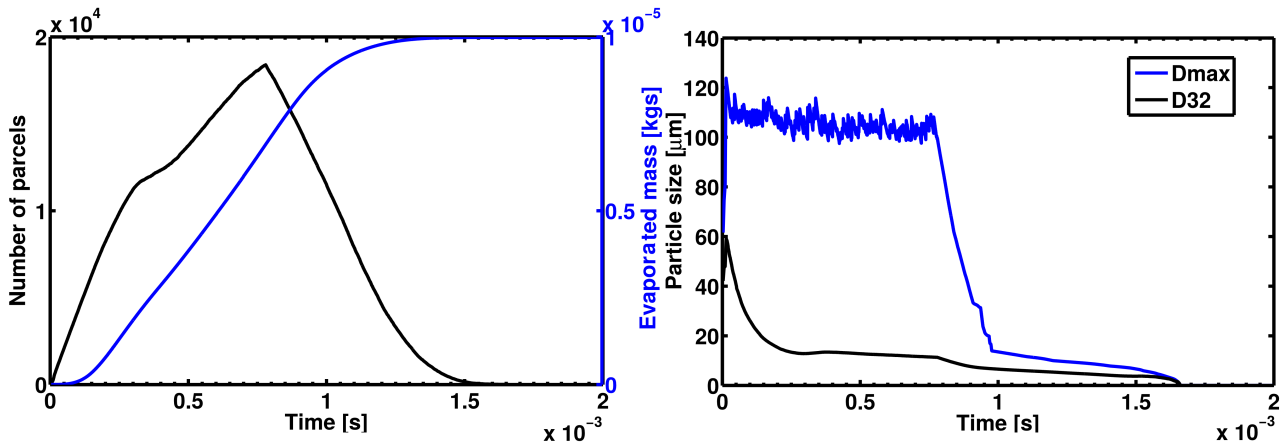


Source: LES (Elaborated by the author). Experiment (ECN, 2018).

### 5.2.2.2 Particle Count and Droplet Size

In addition to the penetration length curves shown above, the plot on the left hand side of Figure 53 shows the number of parcels (black line) and the evaporated fuel mass in kgs (blue line) along simulation time. The number of parcels and evaporated mass steadily increase until end of injection is reached. After that, due to evaporation, parcel count starts to decrease at a constant rate as the fuel mass continues to increase monotonically until all the liquid phase disappears just after 1.5 ms. Moreover, on the right hand side of the same figure, the spray cloud maximum diameter alongside with its Sauter-mean diameter are shown for all simulation duration. The maximum diameter consistently remains above  $100\ \mu\text{m}$  during all injection time, at which point it drastically declines until reaching figures close to the Sauter-mean diameter values at 1 ms.

Figure 53 – Number of parcels/evaporated mass and droplet size along time

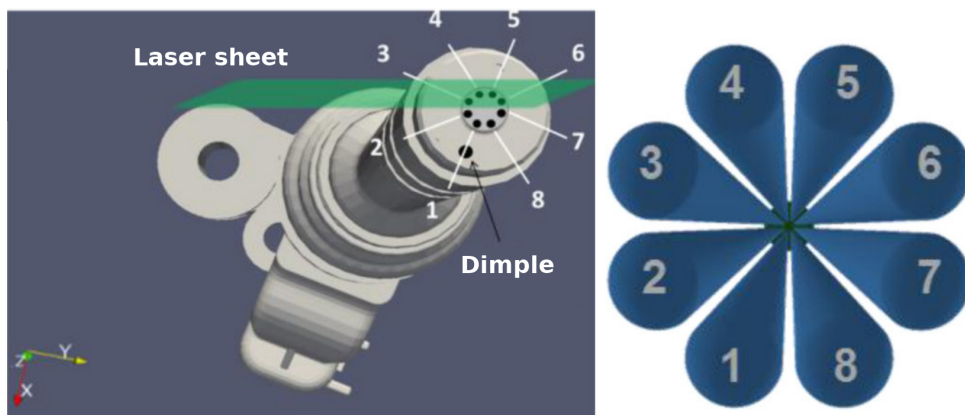


Source: Elaborated by the author.

### 5.2.2.3 Gas Velocity

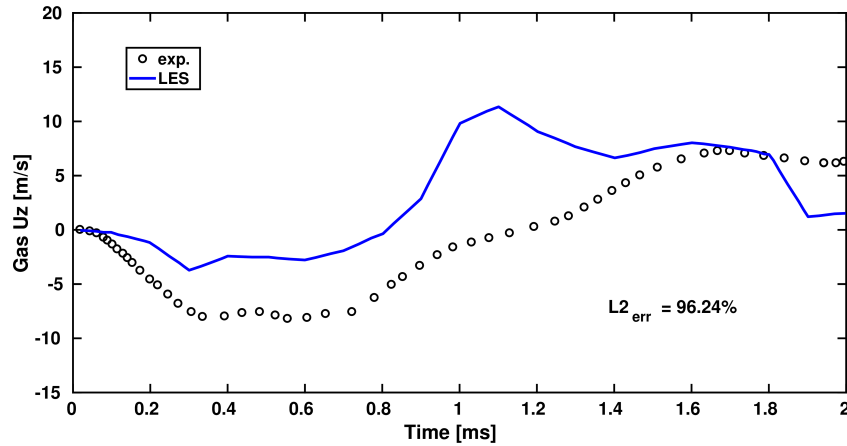
Recent publications presenting experimental results of the gas phase velocity at certain sampling lines from start of injection until 2 ms are provided by Sphicas et al. (2017a) and Sphicas et al. (2017b). The authors performed PIV measurements of the gas axial velocity within a plane crossing in between jets number 2 and 3, as shown in Figure 54. The sampling plane where PIV measurements were made is named as the  $22.5^\circ$  plane, since it bissects the angle formed by the two jets ( $45^\circ$ ), and the sampling line is located at 15 mm below the tip of the injector along the radial direction. Figure 55 shows how the LES results calculated in this work compare to the experimental PIV results from Sphicas et al. (2017b). Since these profiles vary from negative to positive values, only the relative L2 norm error was calculated, showing a very high value of 96.24%, which evidences a large disagreement between simulation and experimental results, especially around 1 ms ASOI. The reason for such disagreement can be explained by the fact that the LES resolution used (0.5 mm) is very low to capture the gaseous velocity within the spray cloud at positions close to the injector.

Figure 54 – Sampling plane at  $22.5^\circ$ , as defined by Sphicas et al. (2017b)



Source: Adapted from Sphicas et al. (2017b).

Figure 55 – Gas axial velocity at 22.5° plane

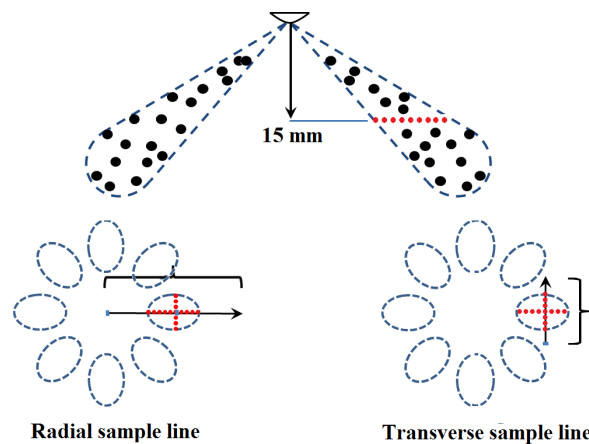


Source: LES (Elaborated by the author). Experiment (SPHICAS et al., 2017b; SPHICAS et al., 2017a).

### 5.2.3 Results at Specific Time-steps

In regard to results at specific time-steps, a few plots concerning droplet velocity and size along sampling lines defined by the ECN community are here discussed. Before these results can be presented, it is important to define the locations where both simulation and experiment results are sampled to be compared. As shown in Figure 56, the sampling lines are located 15 mm downstream from the injector tip, with the radial line going from the core of the spray cloud toward plume direction, whereas the transverse line crosses the middle of the plume perpendicularly to the radial one.

Figure 56 – Definition of sampling lines for comparison between simulation and experiment



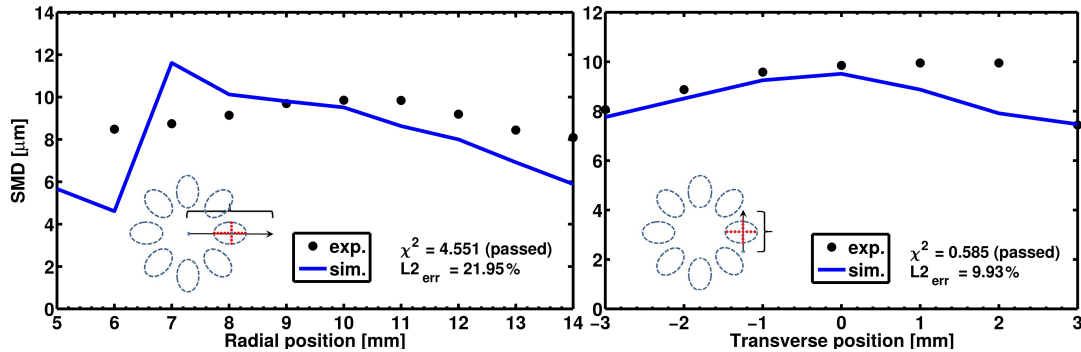
Source: ECN (2018).

#### 5.2.3.1 Droplet Size

The Sauter-mean diameters, in Figure 57 and Figure 58, at positions along the sampling lines defined by the ECN guidelines are shown in comparison to the experimental data measured by Parrish (2014a) for times 0.5 ms and 0.6 ms ASOI (0.3 ms and 0.2 ms before end of injection). The numerical models achieve good performance in predicting the droplet diameters, passing the  $\chi^2$  test with 95% confidence level in all profiles. The relative L2 norm error values are also very low, with a maximum

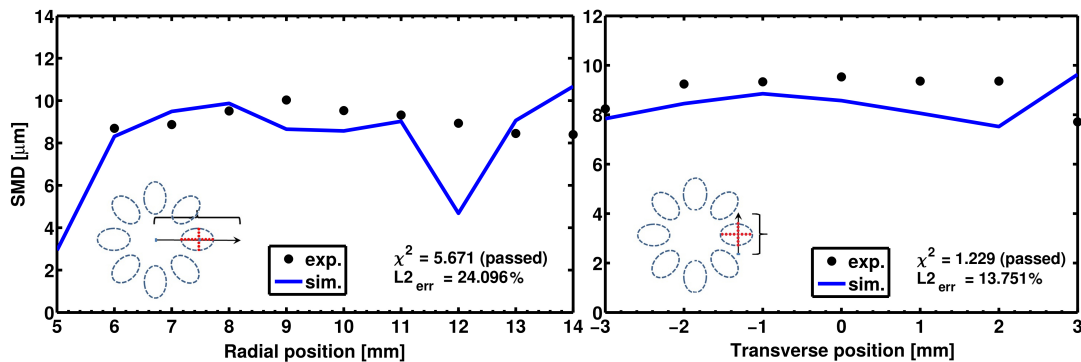
value of 24% for the radial direction at 0.6 ms and a minimum of 9.9% for the transverse direction at 0.5%. Thus, a good agreement was obtained between the LES simulation and the experiment.

Figure 57 – (a) SMD in radial direction at 0.5 ms ASOI, (b) SMD in transverse direction at 0.5 ms ASOI



Source: LES (Elaborated by the author). Experiment (PARRISH, 2014a).

Figure 58 – (a) SMD in radial direction at 0.6 ms ASOI, (b) SMD in transverse direction at 0.6 ms ASOI



Source: LES (Elaborated by the author). Experiment (PARRISH, 2014a).

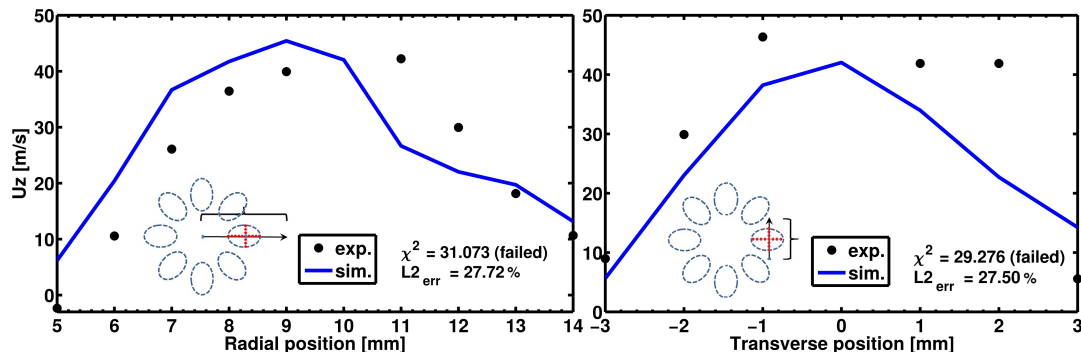
### 5.2.3.2 Droplet Velocity

Finally, using the same sampling lines and for the same times, in Figure 59 and Figure 60 the droplet axial velocities in the simulation are also compared against the measured ones. The simulation results exhibit relative L2 norm errors lower than 30% in both radial and transverse directions, but only results in radial direction at 0.6 ms pass the  $\chi^2$  test with a 95% confidence level.

In both droplet size and velocity profiles shown above, possible reasons for deviation can be attributed to not sufficient high mesh resolution in the dense spray region, wrong model assumptions (for example, all droplet perfectly spherical), and low number of samples in the simulated spray clouds.

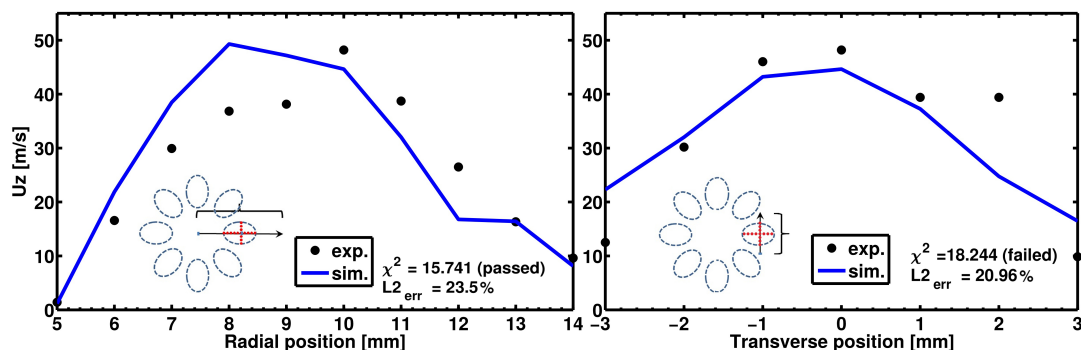


Figure 59 – (a) Axial droplet velocity in radial direction at 0.5 ms ASOI, (b) Axial droplet velocity in transverse direction at 0.5 ms ASOI



Source: LES (Elaborated by the author). Experiment (PARRISH, 2014a).

Figure 60 – (a) Axial droplet velocity in radial direction at 0.6 ms ASOI, (b) Axial droplet velocity in transverse direction at 0.6 ms ASOI



Source: LES (Elaborated by the author). Experiment (PARRISH, 2014a).

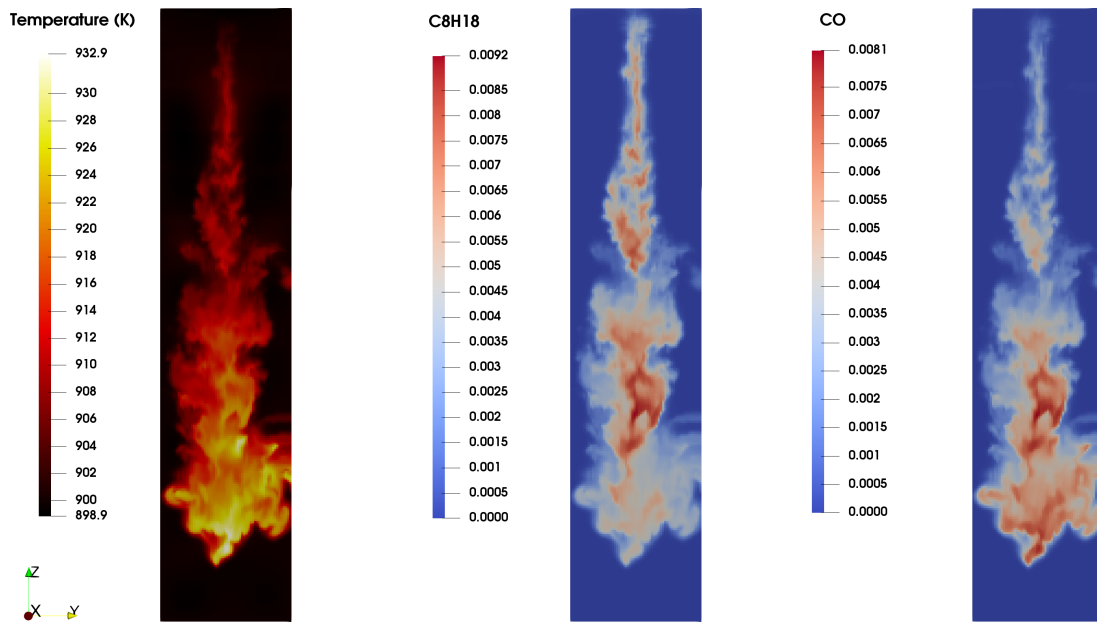
### 5.3 HETEROGENEOUS MIXTURE COMBUSTION STUDY

Next, the results concerning the LES of the ECN “Spray A” condition are presented, this time with the intention to test the combustion packages of the code against experiments. For this condition, three different fuels were used: PRF100 (iso-octane), PRF0 (n-heptane), and PRF60 (blend with 40% n-heptane and 60% iso-octane). For setup information, please refer to Appendix B.2 and the Annex.

#### 5.3.1 Flow Visualization

The simulations of the Spray A reactive injection case were carried out using the Massachusetts Green High Performance Cluster (MGHPCC, 2019). The simulation duration depended on the fuel and the complexity of the reaction mechanism used. The PRF100 case with the reaction mechanism of Zhou et al. (2016) took approximately 7 days to run. The simulated time is directly correlated to the octane number, since it takes more time for ignition to start at increasing concentrations of iso-octane. Therefore, the longest simulation time happens for the octane number of 100 (PRF100), comprehending 10 ms, from which only 3 ms is the injection duration time. In Figure 61, a few plots of temperature and iso-octane/carbon monoxide mass fractions at the central plane for the injection of pure iso-octane are shown. The snapshot is taken at 6 ms, only 2 ms before the start of ignition.

Figure 61 – Temperature and iso-octane/carbon monoxide mass fractions at the central plane for the PRF100 case as ignition starts to unfold at 6 ms

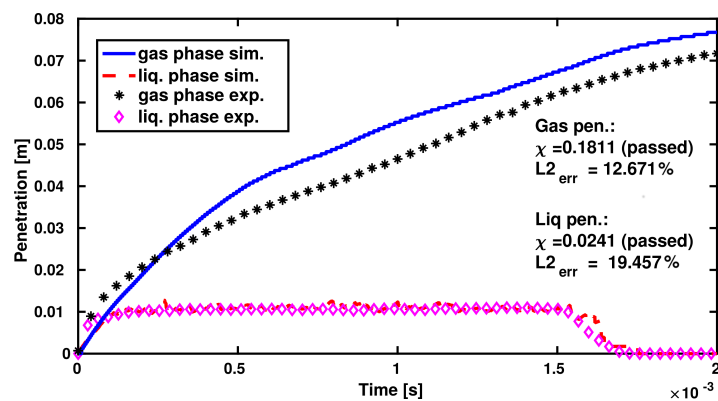


Source: Elaborated by the author.

### 5.3.2 Penetration Length

In Figure 62, the penetration length for both liquid and gas phases of the non-reactive simulation (PRF0 is injected in inert atmosphere) is presented and validated against experimental data. The agreement with the experiment is quite good for the liquid penetration, whereas for the gaseous penetration the simulated results slightly over predict the measured ones after 0.25 ms, passing the  $\chi^2$  test (95% confidence level) with relative L2 norm error of 12.67% (gas) and 19.46% (liquid).

Figure 62 – Penetration length for both liquid and gas phases



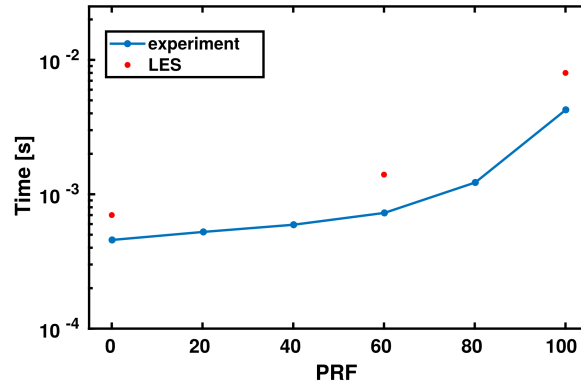
Source: LES (Elaborated by the author). Experiment (ECN, 2018).

### 5.3.3 Ignition Delay

In Figure 63, results concerning the delayed time between start of injection until start of ignition is presented and compared against experimental data. The ignition delay is calculated according to the ECN definition that considers the time at which the rate of increase of maximum temperature is the

highest. As shown, simulated results of ignition delay based on this definition considerably over predict experimental values for both simulations. For pure n-heptane (PRF0), ignition starts at 0.7 ms in the simulation, and at about 0.45 ms in the experiment, whereas for pure iso-octane (PRF100) the ignition delay time is 8 ms, which is almost twice the time needed for the fuel to ignite in the experiment. For the PRF60 fuel, the simulation yields an ignition delay of 1.4 ms, against 0.75 ms from the experiment.

Figure 63 – Ignition delay against experimental data

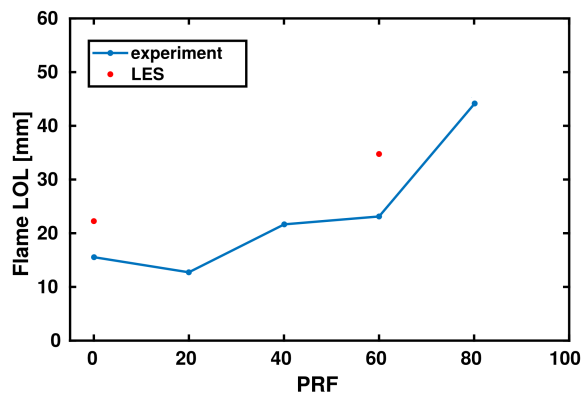


Source: LES (Elaborated by the author). Experiment (PASTOR et al., 2016).

### 5.3.4 Flame Lift-off Length

The flame lift-off length, which according to recent ECN standards is defined as the position from the injector nozzle where the concentration of OH first surpasses 5 % of its maximum concentration, is presented in Figure 64. The flame caused by the injection of pure iso-octane is no longer "lifted", since the ignition starts after the end of injection. However, for octane numbers 0 and 60, a lifted flame is formed and the simulation results are compared to the measured ones. As shown, simulated results again over predict experimental ones, but this time not very considerably. For pure n-heptane, the flame is lifted at 22.25 mm in the simulation, and at about 18 mm in the experiment, whereas for an octane number of 60 the simulated flame lift-off length is 34.75 mm, and about 24 mm in the experiment.

Figure 64 – Flame lift-off length validated against experimental data



Source: LES (Elaborated by the author). Experiment (PASTOR et al., 2016).

## 5.4 GASOLINE DIRECT INJECTION ENGINE STUDY

After testing the engine flow models along with the fuel-spray and combustion packages against experimental results, this section presents the complete GDI engine solver. More information on the setup of these simulations can be found in Appendix B.3 and in the Annex.

In order to test the solver, another case study involving the Darmstadt engine was performed. In this case study, the engine speed is kept constant at about 800 rpm, but this time a reactive mixture of iso-octane and air is provided in the combustion chamber. This mixture is formed by a double injection approach, with the first injection during intake stroke creating an homogeneous fuel-air mixture and the second injection at 76 CAD bTDC with duration of 1.92 CAD generating a stratified condition. The fuel is injected directly into the combustion chamber by means of a pressure swirl atomizer injector mounted on the top of the Darmstadt engine, with total injected mass summing up to 13 mg, from which only 2.9 mg originated from the late injection during the compression stroke. A total of 10 cycles have been calculated from the start of the late injection at 76 CAD bTDC until several crank angle degrees after the total consumption of the fuel present in the combustion chamber. Finally, the spark timing was set to 19 CAD bTDC and the initial flow conditions (velocity, temperature, pressure, etc.) are obtained from the first 10 cold-flow cycles presented in the first section of this chapter. The solution of the combustion process (from spark until 50 CAD aTDC) of 1 cycle of the engine under GDI operation took 5 days of computation using 1 node of the Comet super-computer (SDSC, 2019).

### 5.4.1 Flow Visualization

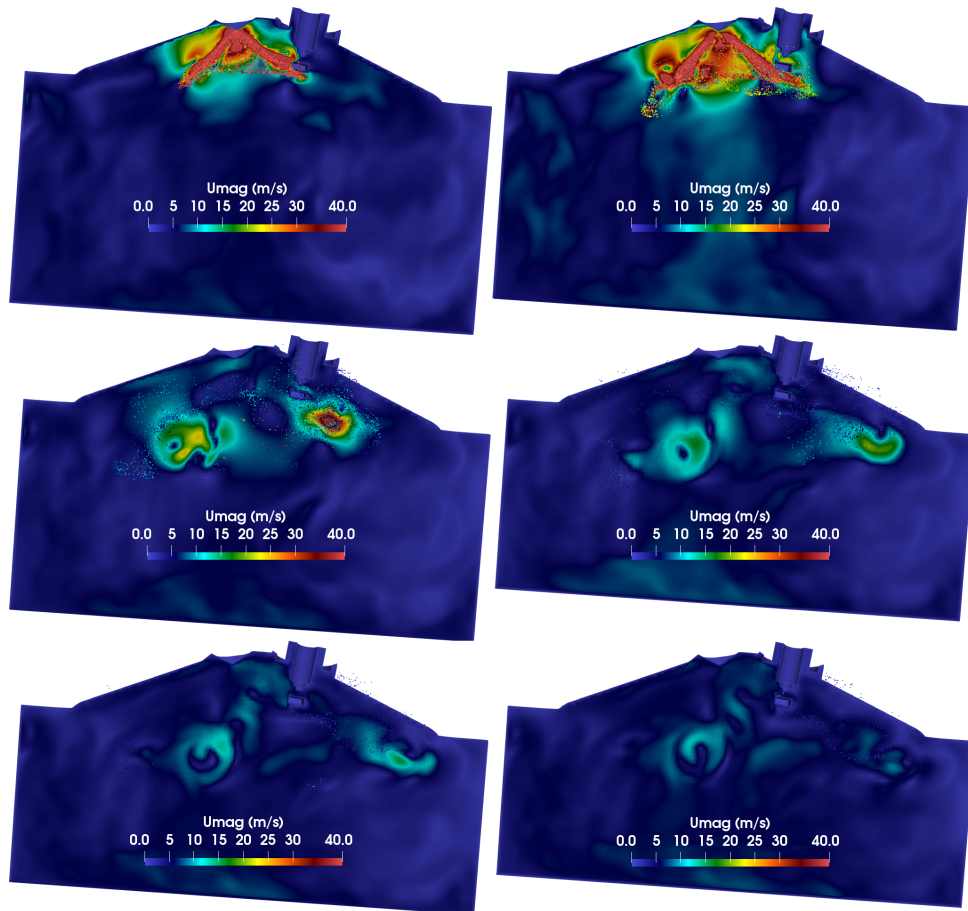
#### 5.4.1.1 Fuel-injection

The next figures show velocity magnitude plots of the GDI engine simulation during fuel injection, starting at 75 CAD bTDC until 60 CAD bTDC, while in Figure 66 the Lagrangian particles colored by velocity are shown together with plots of the velocity vectors of the gas phase, the fuel mass fraction, and the temperature field in the gas phase on the engine central plane. Due to the fuel injection, velocities of over 200 m/s impact the gaseous flow field inside the engine cylinder, causing a considerable increase in the gaseous velocity magnitude. A more detailed analysis of this effect will be presented in the following subsection. The fuel-spray cloud experiences several phenomena during formation of the mixture, such as droplet breakup, heat transfer and evaporation, in a very narrow period of time. At 60 CAD bTDC, only a few parcels are left in the domain.

The spray angle of  $105^\circ$  is intended to direct some of the fuel bulk mass towards the spark location, which is located very close to the injector nozzle position on the cylinder head. Since the variation of the Darmstadt engine simulated in this test case is the spray-guided one, the fuel-spray is guided by itself and the quality of the combustion process at every cycle is very dependent on how much fuel makes it to the vicinity of the spark-plug. Moreover, the high angle of the hollow cone spray formed by the pressure swirl atomizer leaves some considerable amount of gas within the spray cloud, thus inducing some level of flow recirculation, as shown in Figure 66. The recirculation zones along the shear layer between the liquid jet and the air enhances turbulence within the combustion chamber, thus enabling an ignitable mixture to be formed. Only then ignition can be triggered, allowing the flame

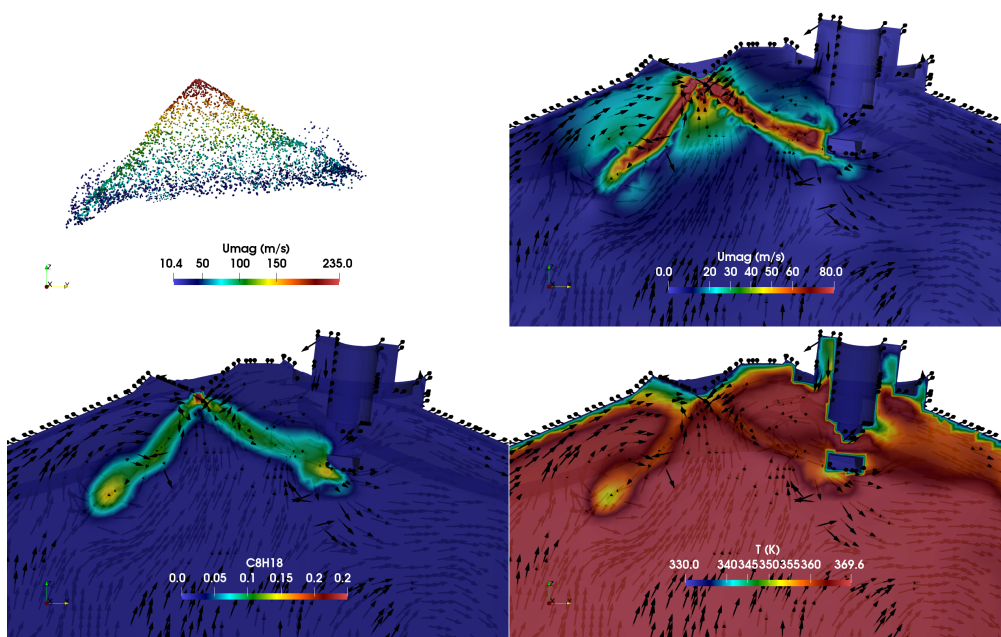
front to propagate towards the piston surface.

Figure 65 – Velocity magnitude during fuel injection from 75 CAD bTDC to 60 CAD bTDC



Source: Elaborated by the author.

Figure 66 – Spray parcels and velocity, fuel mass fraction, and temperature fields

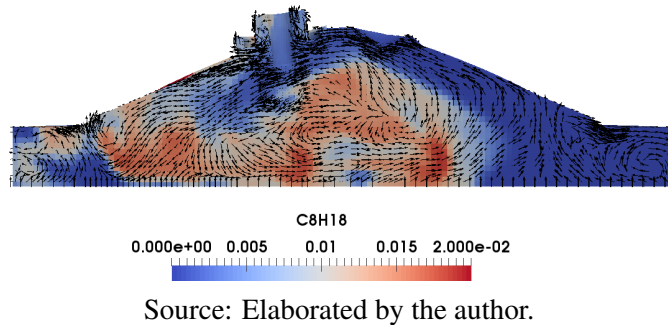


Source: Elaborated by the author.

### 5.4.1.2 Flame propagation

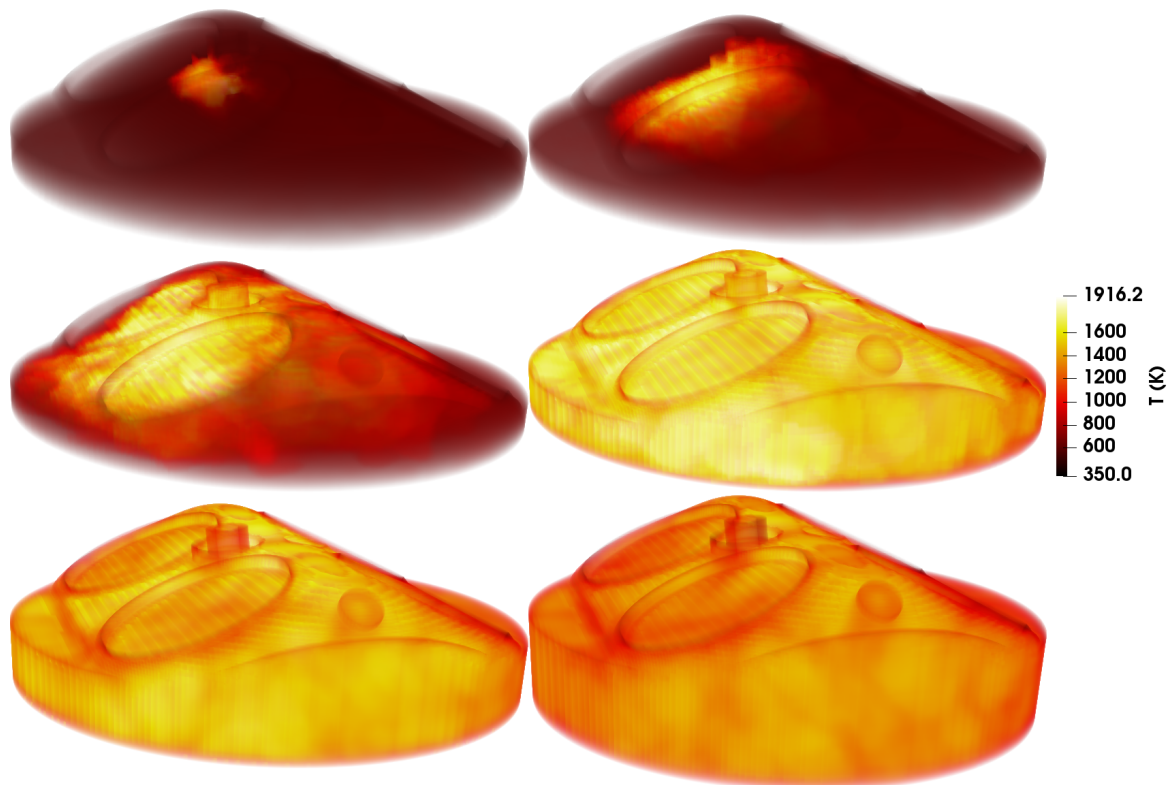
At 19 CAD bTDC ignition is triggered by imposing the temperature of the adiabatic flame in a spherical region between the contacts of the spark-plug. One CAD earlier, at 20 CAD bTDC, the iso-octane mass fraction field together with the velocity vector plot along the midplane is shown in Figure 67, exhibiting not only that the bulk of the gas is going towards the spark region, but also that a considerable amount of fuel is present prior to ignition triggering.

Figure 67 – Iso-octane mass fraction field and velocity vector plot at spark timing



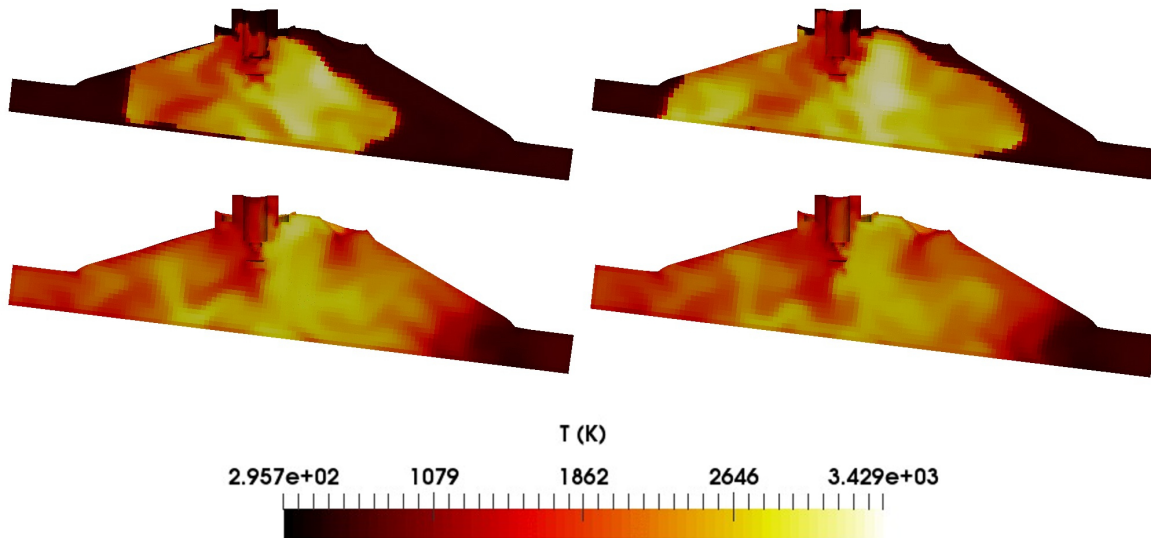
Moreover, Figure 68 presents the temperature field along the midplane from an instant 15 CAD before TDC to another instant 20 CAD after TDC, showing the evolution of the flame front as combustion progresses. As expected, for this particular spark timing, the flame arrives at the piston surface roughly at the same time it approaches TDC.

Figure 68 – Temperature at flame front from 15 CAD bTDC to 20 CAD aTDC



Another interesting observation is that the flame appears to quench as it approaches the region between cylinder head, liner and piston on the right hand side of the plots, due to lower temperatures at the wall in comparison to the flame temperature, as shown in Figure 69. The plot shows the temperature at the flame front from 5 CAD aTDC to 20 CAD aTDC. It can be observed that the flame propagates towards the cylinder liner but it quenches when it finds the gap between the cylinder head and the piston. Flame quenching within engine crevice volumes is a very known and unwanted phenomenon that can lead to undesirable pollutants, such as unburnt hydrocarbon emissions.

Figure 69 – Flame quenching



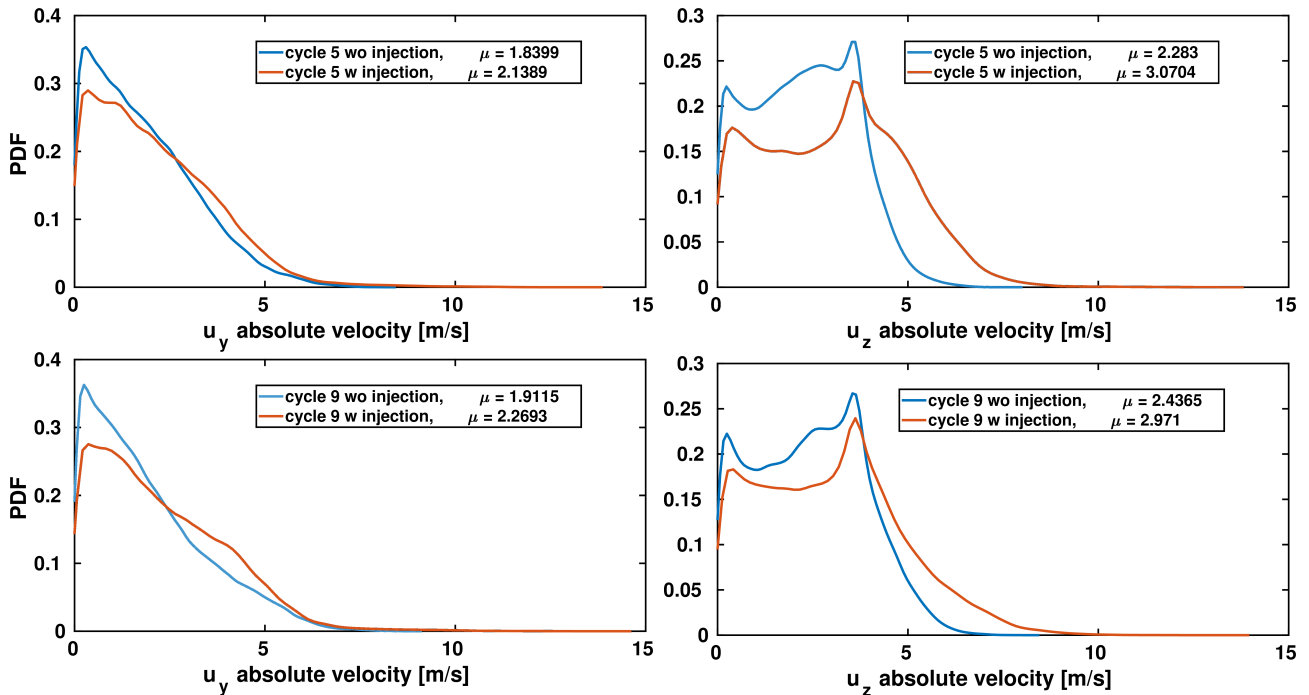
Source: Elaborated by the author.

#### 5.4.2 Impact of Fuel Spray on Turbulence

In order to quantify the impact of the late fuel injection on the in-cylinder velocity field, PDFs of radial ( $y$ ) and axial ( $z$ ) velocity components are calculated for cycles 5 and 9 in two different scenarios. In the first, without the late injection (blue lines), and second, with fuel injection (orange lines), as shown in Figure 69. For cycle 5, the charge within the combustion chamber experiences a mean velocity increase of 16.3% in the radial direction and 34.64% in the axial direction. An increase in the mean velocity is also observed in cycle 9, of 18.48% (radial) and 21.72% (axial).

The velocity increase enforced by the fuel injection has an important effect in enhancing the formation of the mixture prior to spark timing. This is especially important in the case of GDI engines, since the highly transient character of this problem results in a considerable amount of cycle to cycle variation. These cyclic discrepancies, like an over 10% difference in mean axial velocity increase from cycle 5 to 9, can lead to problems if an ignitable mixture cannot be formed in the vicinity of the spark-plug location for some cycles. In this case, misfire or poor burning cycles can occur and compromise the engine performance and fuel efficiency, which counter acts the advantages and the main purpose of investing in the direct injection spark ignited technology. Therefore, new GDI engine designs should aim at reducing such cyclic variations in order to increase the predictability of the combustion process in these engines.

Figure 70 – Comparison of PDFs of the velocity field with and without late injection



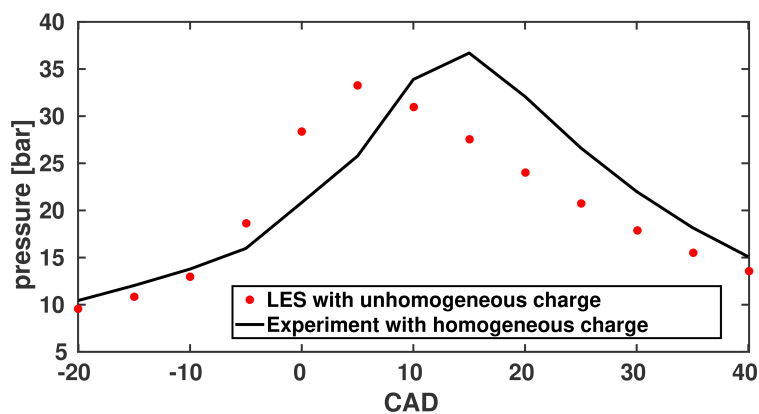
Source: Elaborated by the author.

### 5.4.3 Averaged results along time

In this subsection, some important quantities (pressure, temperature, gas density and mass fraction) averaged spatially and over the calculated fired cycles are presented.

Figure 71 shows the fired in-cylinder pressure trace averaged over 10 cycles from 80 CAD bTDC to 40 CAD aTDC. Unfortunately, due to the lack of experimental measurements of the pressure trace under stratified mixture combustion conditions for this engine, no experimental results are provided for validation. However, the experimental pressure trace of this engine operating with homogeneous mixture is also plotted for reference, but it should be highlighted that this should not be considered as validation. As one can see, the simulated peak pressure with inhomogeneous mixture is slightly higher and considerably late in comparison to the measured results with homogeneous mixture.

Figure 71 – Pressure trace of fired GDI simulation



Source: LES (Elaborated by the author). Experiment (BAUM et al., 2014).

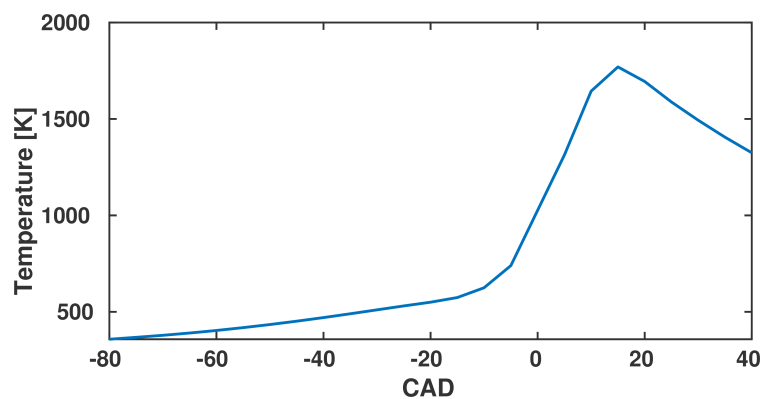
The high peak pressure of 36.7 bar obtained in the LES can be explained by the disregard of



important sources of losses, such as a more detailed wall heat transfer model. Moreover, the late pressure peak, at 15 CAD bTDC, can be explained by the use of an oversimplified single-equation/5-species global reaction mechanism. Because of the complexity of the problem imposed, a detailed or even reduced reaction mechanism could not be applied to these simulations.

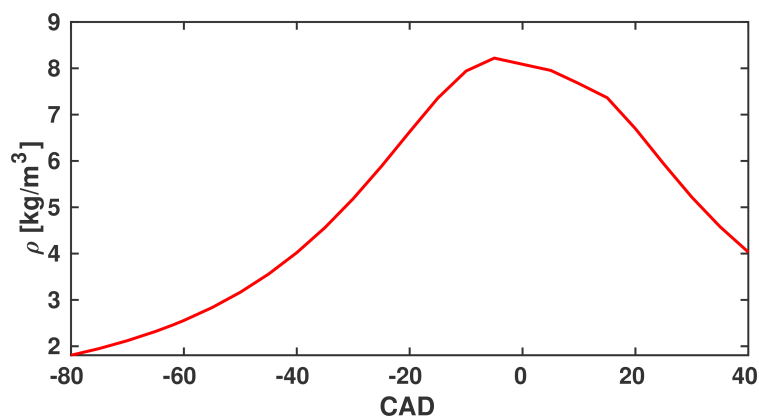
In Figure 72 and Figure 73, the simulated average temperature trace and simulated average gas density trace are shown for the same time range as for the pressure trace presented above. As expected, the temperature inside the combustion chamber peaks at the same time as the pressure does, at 15 CAD aTDC, at a value close to 1800 K. The gas density, however, peaks at a value of  $8 \text{ kg/m}^3$  around the time of 5 CAD bTDC. Low gas density values are obtained in the wake of the flame, which pushes the average gas density down as the flame propagates further.

Figure 72 – Temperature trace of fired GDI simulation



Source: Elaborated by the author.

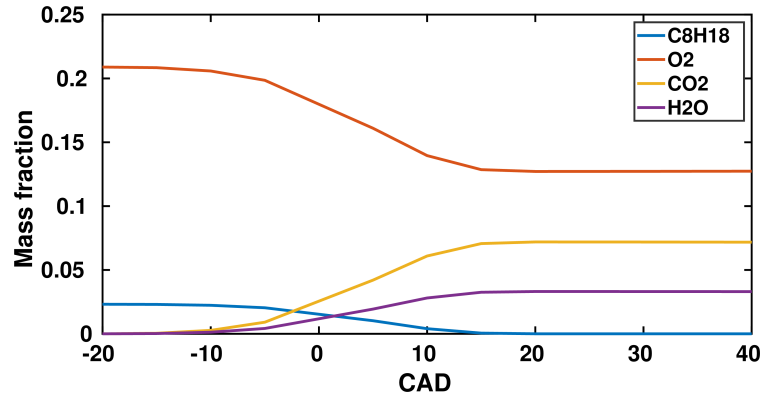
Figure 73 – Gas density trace of fired GDI simulation



Source: Elaborated by the author.

Finally, Figure 74 shows some of the species mass fractions considered in the simulation (iso-octane, oxygen, carbon dioxide and water vapour) along time. As expected, the consumption of fuel and oxygen starts just after ignition is triggered at 19 CAD bTDC, at the same time products carbon dioxide and water vapour start to be formed. The rate of consumption of oxygen, as well as the rate of formation of  $CO_2$  hit peak value in the time range between TDC and 10 CAD bTDC. From 15 CAD aTDC on only a trace amount of fuel is present in the domain.

Figure 74 – Species mass fraction concentrations along time



Source: Elaborated by the author.

In summary, the content presented in this chapter until this point comprehended the testing of all models developed to study GDI engine flows and related phenomena, and the development of the complete open-source GDI engine solver, using the so called conventional computational method (CFD). In order to close this chapter, the next section will present a study on the use of a novel method, i.e. artificial neural networks, to model the behavior of the small scale turbulence in GDI engines, using the dataset generated from the CFD simulations.

## 5.5 ICE NETWORK STUDY

In this last section, the results obtained by the OFICENet are presented. As described in chapter 3, section 3.4, a 4-layer fully-connected neural network using the Keras deep learning framework was employed.

### 5.5.1 Training

A summary of the training log of the network is shown in the appendix, in Frame 2. The dataset used here was obtained from engine simulation data from several different cycles at the bottom dead center, and it has been split between a training and a test set (sometimes also called validation set) with an 80%/20% proportion, which means that 80% of the data is used to train the neural network and the remaining 20% information is used to test the capability of the ANN in making a prediction from data never seen during training. The training has been carried out for several runs of 1000 Epochs each. One Epoch means a complete iteration within the artificial neural network, forward propagating the information from the input layer throughout the hidden layers until the output, calculating the cost function, then back propagating the information for the calculation of the derivatives and running the Adam optimization function using the mean absolute error as metric. The weights calculated by the ANN in a particular Epoch are then used as initial values for the next iteration.

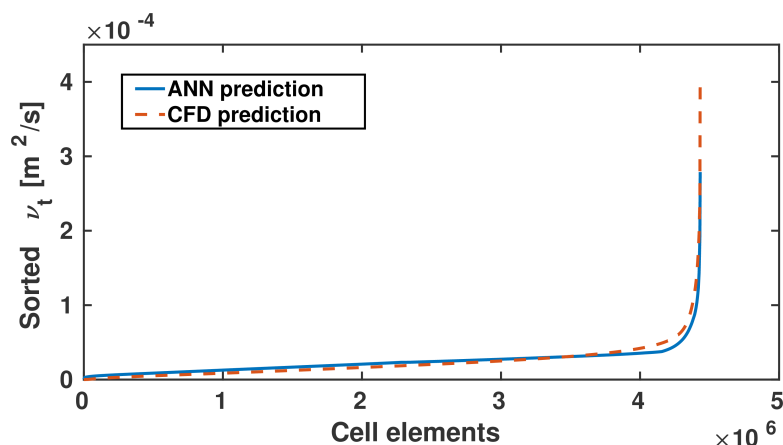
The log shown in Frame 2 refers to the last run executed, obtaining a minimum training loss of 0.001424 and a validation loss of 0.001695, taking approximately one day to run the 1,421 parameters ANN in a single 2.2-GHz Core of an i7-8750H processor. The number of parameters had to be limited to only slightly more than one thousand due to the limited computational resource used for the ANN

training, and also due to characteristics of fully-connected networks, in which weights calculated by each neuron may vanish or “explode” (increase rapidly) if too many layers are used. Moreover, it is also recommended to keep the number of neurons in each layer to a number not much higher than the number of features in the first layer (NG, 2017). Therefore, since a number of 6 features were used, a total of 20 neurons were given to each layer.

### 5.5.2 Results

Figure 75 presents an example of the potential of the ANN in using the weights obtained by training to make a prediction of an important property related to the small turbulent scales. This prediction is made for a new engine cycle not seen during training and its results are shown for a plot of the turbulent kinematic viscosity by the number of cells at bottom dead center. The result (y-axis) values were sorted in ascending order with the purpose of facilitating the reading of the data, because otherwise it is very hard to distinguish between the CFD (ground truth) and the ANN noisy results. As it can be seen, the ANN model seems to underpredict the peak value obtained by the CFD simulation. The CFD result predicts a peak value of  $4.06 \times 10^{-4} \text{ m}^2/\text{s}$ , whereas the ANN predicts a peak value of  $2.78 \times 10^{-4} \text{ m}^2/\text{s}$ , which translates into a 31% difference. Moreover, the ANN predicted values seem to be biased towards the cell averaged value calculated by the CFD model, slightly overpredicting the results of the first 3 million cells and then slightly underpredicting after that. This situation could be corrected by training the network with more data and/or proposing a new network architecture, such as a convolutional neural network. The latter option would even allow the extra advantage of deepening the network even further, using one of the well established CNN architectures in the literature (LECUN et al., 1998; KRIZHEVSKY; SUTSKEVER; HINTON, 2016; SZEGEDY et al., 2015; SIMONYAN; ZISSERMAN, 2015), and thus increasing the number of parameters to be learned from a few thousands to hundreds of thousands or even millions, without the need for a substantial increase in computational power.

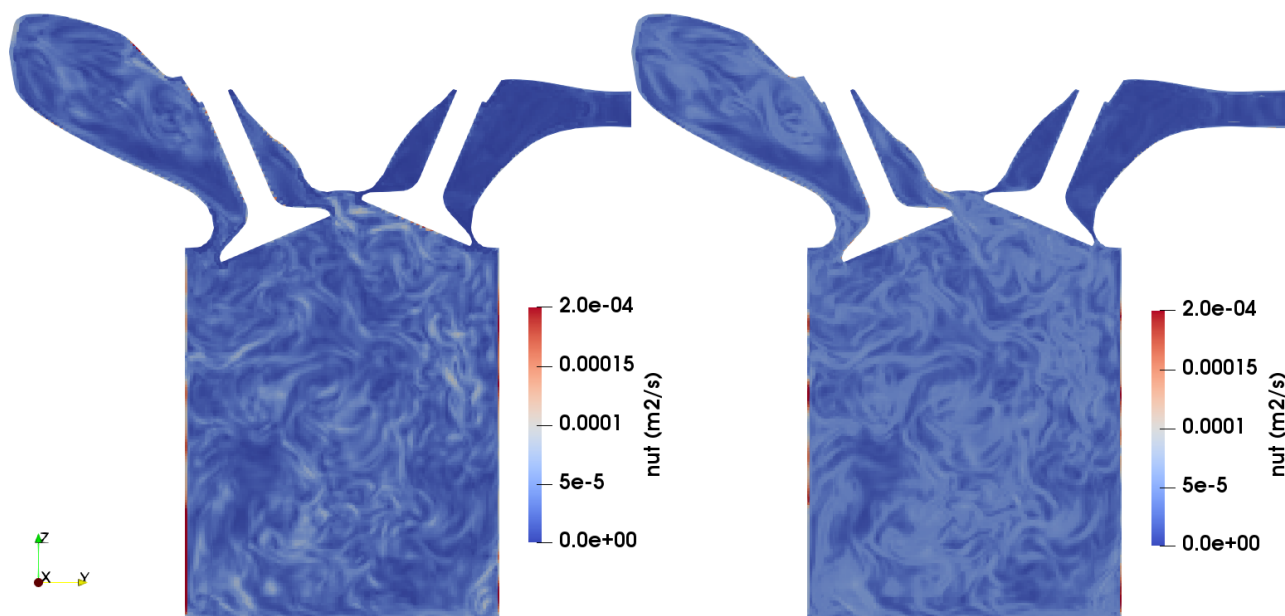
Figure 75 – Calculation of turbulent kinematic viscosity using CFD and deep learning



Source: Elaborated by the author.

Next, Figure 76 provides a visual aid to understand the aforementioned results, showing a plot of the turbulent kinematic viscosity on the plane over the valves at bottom dead center with CFD prediction on the left and ANN prediction on the right.

Figure 76 – Prediction from CFD (left) against prediction from deep neural network (right)



Source: Elaborated by the author.

The plot above confirms what has been observed in the previous graph, i.e., an apparent bias of the ANN prediction towards the cell averaged result from the CFD calculation. Although the obtained results are very preliminary and the errors still appear to be considerable, the ANN model can already capture structures that are very similar to the ones calculated using the conventional CFD model, despite the different ranges of magnitude. These similar structures also validate the choice of the gradient of the velocity field as input features, thus in the future, similar quantities (such as the Reynolds stress tensor invariants) can be added in order to improve the complexity of the model.

Furthermore, these results also show the potential of the use of deep learning coupled with CFD codes in order to improve or accelerate CFD calculations. Provided more training is performed or a more complex model is employed, the ANN predictions of small scale turbulent quantities may become as good as the ones obtained by the CFD code. In this case, the weights from a neural net trained with a huge engine dataset using a fine mesh can be later used to support turbulent models of CFD simulations in coarser meshes, with no or very little additional training. Thus, the same level of accuracy can be obtained with less computational resources, due to the “help” provided by the learned weights using the fine mesh data. The learned weights can be easily implemented in the CFD code to make the predictions of the turbulent kinematic viscosity values by shutting off the conventional subgrid model (Smagorinsky in this particular case), replacing its output with the ANN output by means of simple matrix multiplications (as explained in chapter 3).

These results conclude the findings obtained from the numerical investigation of the Darmstadt optical engine performed in this work and the discussions achieved from it. In the next and last chapter, these results will be summarized and conclusions will be drawn from both conventional CFD and novel ANN strategies.

## 6 CONCLUSIONS

In this doctoral thesis, a thorough study on computational methods applied to internal combustion flows has been performed, using both conventional CFD methods, and a novel method, based on the deep learning technology. Three different case studies have been chosen to test the models developed here, namely the Darmstadt optical engine and the ECN "Spray G"/"Spray A" injection cases, due to the wide range of high quality experimental results available, such as optical measurement techniques like PIV, Mie scattering, shadowgraphy, among others.

### 6.1 RESEARCH SUMMARY

The research focused primarily on questions related to the complex turbulent flow field induced by moving boundaries (valves and piston) operating at very high speeds, for which the Darmstadt engine was the best candidate, and the complex multiphase flow initiated by the injection of gasoline followed by the combustion of the subsequent reactive environment, for which the ECN spray cases were suitable. Secondly, the thesis focused on the contributions of this work to the literature, which comprehended (1) the creation of a free and open-source CFD code for the simulation of internal combustion engines and related phenomena, making the research of these accessible to more people, especially where resources are scarce, and (2) the proposal of a deep neural network model (OFICEnet) for predicting properties of the small turbulent scales in order to show the potential of deep learning as a method to improve and accelerate CFD codes. The originality of the first contribution, the OFICE code, lies in the fact that the simulation of reactive flows associated to IC engines is entirely dependent on commercial codes, which are not easily accessible due to the high associated license costs. Therefore, to the author's knowledge, the OFICE code is the first GPLv3 licensed CFD software capable of performing complete DISI engine simulations. In regard to the second contribution, even though a few researchers have successfully applied deep learning techniques to fluid dynamics simulations in recent publications, the author is unaware of the use of such techniques to predict flow properties in internal combustion engines.

For the investigation on the Darmstadt engine's internal flow, an advanced turbulence modeling approach, namely the large-eddy simulation Smagorinsky model, was employed, in contrast to the conventional RANS techniques largely employed by both industry and academia for modeling engine flows. The LES approach was then coupled to a moving grid strategy based on the solution of the Laplace equation, thus allowing the movement of piston and valves. The mesh deformation generated by these moving boundaries was dealt by the mapping approach, i.e., generating multiple grids for an entire engine cycle and mapping the results from one mesh to the subsequent one before mesh deformation starts to become a problem. These combined strategies were then extensively applied to the Darmstadt optical engine case, and the 20 LES cycles were compared against the available experimental results. As good results were obtained, backed by statistical techniques (chi-square, relative L2-norm error and PDFs), the research turned into the investigation of the characteristics of the large scale turbulence behavior, like the tumble motion. For that, the vortex center tracking algorithm

of Graftieaux, Michard and Grosjean (2001) was applied to provide some qualitative information about the formation and extinction of the tumble motion along the compression stroke. Moreover, the modified two point velocity correlation proposed by Janas (2017) was then employed to fill the gaps left by the previous analysis with a more quantitative approach, from which a length scale for the turbulent coherent structures was derived. It was shown that towards the end of the compression stroke, at TDC, a small tumble vortex is still present in the combustion chamber, even though its intensity is much reduced from the instant the intake valves close. Finally, the LES quality criterion proposed by Pope (2011) was applied to quantify how much the turbulent kinetic energy is resolved and how much of it is modeled. As shown, for almost the entire cycle, the averaged evaluation of Pope's quality criterion remains below the recommended 20% threshold for the resolution used (base cell size of 0.7 mm and minimum cell size of 0.125 mm).

The next focus became testing the fuel-spray methodology intended to model the fuel injection within the engine model using a Lagrangian-Eulerian framework in both non-reactive and reactive conditions. Thus, simulations of the ECN "Spray G" were performed to evaluate the models in a non-reactive environment. Once more, LES results were compared against experimental values when measurements were available, which was the case for the metrics of spray liquid/gas penetration length, gas axial velocity, and droplet size/velocity distribution. When possible, the accuracy of the LES results was evaluated using standard statistical tools (chi-square and relative L2-norm error). While the model passed the chi-square test with a confidence level of 95% for most metrics tested, it did not show optimal performance when challenged to predict the velocity field, especially in the gas phase, in the region close to the injector tip. These findings corroborate the necessity of improving fuel-spray models in the dense spray region, where, unfortunately, experimental results are hard to obtain, and may also be evidence that a greater resolution is needed close to the injector; thus adaptive mesh refinement (AMR) should be considered in the future.

Regarding the performance of the fuel-spray models in a reactive environment, further LES of the ECN "Spray A" was conducted. The simulated test case is actually a modified version of the original "Spray A" case, since the original considers only a diesel surrogate as fuel, whereas the former considers both gasoline and diesel surrogates (blends of iso-octane and n-heptane). Due to the complexity of the problem, the reduced reaction mechanism of Zhou et al. (2016) considering 43 species in 144 different reactions was employed to simulate three different iso-octane/n-heptane fuel blends with iso-octane proportions 0%, 60%, and 100% (here denoted as PRF0, PRF60, and PRF100). As a first test, the simulated liquid and gaseous penetration curves for the PRF0 fuel in non-reactive environment were compared against the available ECN experimental results, showing good agreement between LES and measurements. Regarding the tests carried out in the reactive environment, the experimental results of ignition delay and flame lift-off length measured by Pastor et al. (2016) were used. The numerical model showed a tendency to overpredict ignition delay times considerably, especially for the case with pure iso-octane fuel. For the flame lift-off length, the model also overpredicts the measured results for the two possible conditions considered (ignition delay for PRF100 is longer than the injection duration, thus this flame is not lifted), but less considerably than in the case of the ignition delay. The disagreements in ignition delay can be explained by the

lack of complexity of the reaction mechanism used, and also by the fact that kinetics models tend to predict slower ignition starts unless a great deal of constant tuning is done to the combustion/reaction mechanism models (WARNATZ; MAAS; DIBBLE, 2006). However, this kind of adjustment is not the purpose of this work, and this would not be a reasonable approach for a code that aims to be applied to a wide range of applications. Finally, the disagreement in flame lift-off length calculations can also be explained by simplifications to the reaction mechanism, as well as by the lack of resolution within the dense spray region.

Furthermore, all numerical models discussed above were implemented into the final version of the OFICE solver and results regarding the average LES fired engine cycles were presented, using the first 10 consecutive cycles from the cold-flow simulation as initial condition. The ability of the code to model DISI engines was demonstrated with some qualitative and quantitative plots of the velocity field, the flame temperature, and the averaged pressure, temperature, and species mass fractions along time. Unfortunately, since there are no available experimental results of the Darmstadt engine operating under DISI condition, the pressure trace could not be compared to measurements conducted under the same configuration. However, in order to give the reader some point of reference, the experimental pressure trace under homogeneous condition was also plotted. Even though it is hard to claim with confidence because of the lack of validation data, the simulated pressure trace seems to develop slowly, producing a high peak pressure around 15 CAD aTDC. This behavior can be explained by the use of an overly simple single reaction mechanism comprised of only 5 species (fuel, oxygen, carbon dioxide, water, and nitrogen) due to computational limitations. In addition, PDFs of components of the velocity field in the radial and axial directions were evaluated in both cold and fired simulations, showing that the fuel injection has substantial effect on the flow field.

Finally, a study concerning the training of an artificial neural network (OFICenet) with the engine simulation data produced in this work was presented. This study has shown the potential of deep learning in making CFD like predictions of the small scale turbulent behavior (turbulent kinematic viscosity), indicating how this could be leveraged to accelerate CFD codes. If the predictive capability of the OFICenet is improved through more training and assigning it a more complex model, one can take advantage of the OFICenet trained weights using fine mesh simulation data to produce high quality results in coarser meshes with less powerful and more affordable computational resources.

## 6.2 FUTURE WORK

Despite answering some of the imposed questions on the possibility of developing a validated full open-source DISI engine solver, as well as using the power of deep learning to generate CFD-like calculations of small scale turbulence properties, this research has also left some open questions that could be addressed in future investigations.

Regarding the conventional modeling approach, although the OFICE code has passed some important tests when validated against experimental results under motored condition, there is still a lot of room to improve the fuel-spray models, especially in regard to combustion and reaction kinetics modeling. In a future work, finer meshes and AMR should be used to address the lack of resolution within the dense spray region, and an investigation on the use of other reduced mechanisms, maybe

with higher complexity, should be conducted in order to assess the disagreements seen in ignition delay and flame lift-off length (especially if the code is intended to be used to model diesel engines). Moreover, it is necessary to find a database to test the full solver with experimental DISI engine data under fired condition.

Regarding the novel modeling approach, the current deficiencies of the OFICEnet model must be overcome with more training, perhaps using other engine simulation data, and also with more complex ANN models and architectures. The most prominent first attempt would be the use of a convolutional neural net (CNN) instead of the fully-connected one used so far because CNN nets are recommended for pattern recognition. Furthermore, the potential of such nets have already been shown for CFD like predictions (GUO; LI; IORIO, 2016; TOMPSON et al., 2017). After that, the next phase would be to extend the predictive capability of the OFICEnet with the development of a data-driven framework to permit interactions with new phenomena, such as multiphase and reacting flows. The goal then will be to train the neural network in much finer meshes, so that the learned weights can be later used to generate predictions of the subgrid scale turbulence behavior in CFD simulations using coarser meshes, thus achieving the same level of accuracy but with a smaller computational cost. In that case, additional training with other engines will be necessary so that the model is not biased towards the Darmstadt engine and new input features will also be considered.

In conclusion, this work has also contributed to the creation of the ICEnet consortium at the University of Massachusetts Amherst. This consortium was formed by means of a partnership between academia and industry, with the intention to develop software to model the phenomena of turbulence, multiphase and reacting flows employing both physics-driven (CFD) and data-driven (ANN) approaches. The OFICE code was chosen for the CFD approach, with its solver intended to run on high performance computers based on both CPUs and GPUs (graphical processing units) to generate internal flow data from several different engines. This vast dataset is then used to train an improved version of the ANN approach presented in this work (OFICEnet), and the trained weights are incorporated into a final code combining both CFD and ANN frameworks. Consequently, both contributions outlined here, the open-source GDI engine CFD solver and the ICEnet network, can be leveraged in the future, and hopefully answer the open questions left by the current work.



## REFERENCES

- ADACHI, M.; MCDONELL, V. G.; TANAKA, D.; SENDA, J.; FUJIMOTO, H. Characterization of fuel vapor concentration inside a flash boiling Spray. **SAE Technical Paper**, n. 970871, 1997.
- AMELANG, S. **Clean energy wire: renewables cover about 100power use for first time ever**. 2018. Available at: <<https://www.cleanenergywire.org/news/renewables-cover-about-100-german-power-use-first-time-ever>>. Access on: 30 Jun 2019.
- AMSDEN, A. A. **KIVA-3V: a block-structured KIVA program for engines with vertical or canted valves**. 1997. Los Alamos National Laboratory Report LA-11560-MS. Available at: <<https://digital.library.unt.edu/ark:/67531/metadc693255/>> Access on: 30 Jul 2019.
- AMSDEN, A. A.; BUTLER, T. D.; O'ROURKE, P. J.; RAMSHAW, J. D. KIVA-A comprehensive model for 2-D and 3-D engine simulations. In: SAE INTERNATIONAL CONGRESS AND EXPOSITION, 94, 1985, Detroit. In: **SAE International**. Detroit: SAE, 1985. p. 1–20.
- AMSDEN, A. A.; O'ROURKE, P. O.; BUTTLER, T. D. **KIVA-II: a computer program for chemically reactive flows**. 1989. Los Alamos National Laboratory Report LA-11560-MS. Available at: <<https://www.osti.gov/biblio/6228444-kiva-ii-computer-program-chemically-reactive-flows-sprays>> Access on: 30 Jul 2019.
- AMSDEN, D. C.; AMSDEN, A. A. The kiva story: a paradigm of technology transfer. **IEEE Transactions on Professional Communication**, v. 36, n. 4, p. 190 – 195, 1993.
- ANDO, K.; TAKAMAEDA-YAMAZAKI, S.; IKEBE, M.; ASAI, T.; MOTOMURA, M. A multithreaded cgra for convolutional neural network processing. **Circuits and Systems**, v. 08, p. 149–170, 2017.
- ANL. **DNS of forced- and auto-ignition in spherical and engine-like geometries**. 2018. Available at: <<https://www.alcf.anl.gov/projects/dns-forced-and-auto-ignition-spherical-and-engine-geometries>>. Access on: 30 Jun 2019.
- ARCOUMANIS, C.; GAVAISES, M.; FRENCH, B. Effect of fuel injection process on the structure of diesel Sprays. **SAE Technical Paper**, n. 970799, 1997.
- ARCOUMANIS, C.; WHITELAW, J. H. Fluid mechanics of internal combustion engines—a review. **Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science**, v. 201, n. 1, p. 57–74, 1987.
- ATAK, M. **The taylor-green vortex**. 2019. Available at: <<https://www.flexi-project.org/?projects=taylor-green-vortex>>. Access on: 30 Jun 2019.
- ATKINSON, J. **Gas engine**. US 367496A, 02 aug. 1887, 02 aug. 1904. Available at: <<https://patents.google.com/patent/US367496>>. Access on: 30 Jun 2019.
- ATSDR. **Chemical and physical information**. 2018. Available at: <<https://www.atsdr.cdc.gov/toxprofiles/tp75-c3.pdf>>. Access on: 30 Jun 2019.
- AVBP. **The AVBP project**. 2018. Available at: <<http://www.cerfac.fr/avbp7x/>>. Access on: 30 Jun 2019.

- BALDWIN, E.; GROVER JR., R.; PARRISH, S.; DUKE, D.; MATUSIK, K.; POWELL, C.; KASTEN-GREN, A.; SCHMIDT, D. String flash-boiling in gasoline direct injection simulations with transient needle motion. **International Journal of Multiphase Flow**, v. 87, p. 90–101, 2016.
- BARITAUD, T.; DUGLOS, J.; FUSCO, A. Modeling turbulent combustion and pollutant formation in stratified charge si engines. **Symposium (International) on Combustion**, v. 26, n. 2, p. 2627 – 2635, 1996. ISSN 0082-0784.
- BASSHUYSSEN, R.; SCHÄFER, F. **Handbuch verbrennungsmotor**. Wiesbaden, Germany: Springer, 2002.
- BAUM, E.; PETERSON, B.; BÖHM, B.; DREIZLER, A. On the validation of les applied to internal combustion engine flows: Part 1: Comprehensive experimental database. **Flow, Turbulence and Combustion**, v. 92, n. 1, p. 269–297, Jan 2014.
- BAUM, E.; PETERSON, B.; SURMANN, C.; MICHAELIS, D.; BÖHM, B.; DREIZLER, A. Investigation of the 3d flow field in an ic engine using tomographic piv. **Proceedings of the combustion institute**, v. 34, n. 2, p. 2903 – 2910, 2013. ISSN 1540-7489.
- BAUMGARTEN, C. **Mixture formation in internal combustion engines**. Hannover: Springer, 2005.
- BEALE, J. C.; REITZ, R. D. Modeling spray atomization with the Kelvin-Helmholtz/Rayleigh-Taylor hybrid model. **Atomization and Sprays**, v. 9, p. 623–650, 1999.
- BODE, J.; SCHORR, J.; KRÜGER, C.; DREIZLER, A.; BÖHM, B. Influence of three-dimensional in-cylinder flows on cycle-to-cycle variations in a fired stratified disi engine measured by time-resolved dual-plane piv. **Proceedings of the Combustion Institute**, v. 36, n. 3, p. 3477 – 3485, 2017. ISSN 1540-7489.
- BOUSSINESQ, J. **Essai sur la théorie des eaux courantes**. 1877. Thesis (PhD in Mechanical Engineering) - Académie des Sciences de L'Institut National de France, Paris, France. 64 p. Available at: <<https://gallica.bnf.fr/ark:/12148/bpt6k56673076.texteImage>>. Access on: 30 Jun 2019.
- BOYCE, W. E.; DIPRIMA, R. C. **Elementary differential equations and boundary value problems**. Hoboken, NJ: Wiley, 2005.
- BUCHAL, C.; KARL, H.-D.; SINN, H.-W. Kohlemotoren, windmotoren und dieselmotoren: was zeigt die co2 -bilanz? **ifo Schnelldienst**, v. 72, n. 8, p. 40 – 54, 2019. Available at: <<https://www.ifo.de/DocDL/sd-2019-08-sinn-karl-buchal-motoren-2019-04-25.pdf>>. Access on: 30 Jun 2019.
- CHEN, L.; LIN, J.; SUN, F.; WU, C. Efficiency of an atkinson engine at maximum power density. **Energy Conversion and Management**, v. 39, n. 3, p. 337 – 341, 1998. ISSN 0196-8904.
- CHOLLET, F. **Keras**. 2015. Available at: <<https://blog.keras.io/author/francois-chollet.html>>. Access on: 30 Jun 2019.
- D'ADAMO, A.; BREDI, S.; FONTANESI, S.; IRIMESCU, A.; MEROLA, S. S.; TORNATORE, C. A rans knock model to predict the statistical occurrence of engine knock. **Applied Energy**, v. 191, p. 251 – 263, 2017. ISSN 0306-2619.
- DAIMLER. **Daimler: engines, carburetors, radiators and transmissions (1883-1901)**. 2018. Available at: <<http://media.daimler.com/marsMediaSite/en/instance/ko/Daimler-engines-carburetors-radiators-and-transmissions-1883-1901.xhtml?oid=9274037>>. Access on: 30 Jun 2019.

- DATE, A. W. **Analytic combustion: with thermodynamics, chemical kinetics and mass transfer**. Cambridge, England: Cambridge University Press,, 2011.
- DIESEL, R. **Diesel**. US 542846A, 16 jul. 1895. Available at: <<https://patents.google.com/patent/US542846A/en>>. Access on: 30 Jun 2019.
- ECN. **Engine combustion network**. 2018. Available at: <<http://www.sandia.gov/ecn>>. Access on: 30 Jun 2019.
- EIA. **Annual energy outlook 2018**. 2018. Available at: <<https://www.eia.gov/outlooks/aeo/pdf/AEO2018.pdf>>. Access on: 30 Jun 2019.
- EIA. **US energy information admnistration: Michigan state profile and energy estimates**. 2019. Available at: <<https://www.eia.gov/state/?sid=MI#tabs-4>>. Access on: 30 Jun 2019.
- FERZIGER, J. H.; PERIC, M. **Computational methods for fluid dynamics**. Berlin Heidelberg: Springer Verlag, 2002.
- FREUDENHAMMER, D.; PETERSON, B.; DING, C.-P.; BÖHM, B.; GRUNDMANN, S. The influence of cylinder head geometry variations on the volumetric intake flow captured by magnetic resonance velocimetry. **SAE International Journal of Engines**, SAE International, v. 8, p. 1826–1836, 04 2015.
- GERMANO, M. Turbulence: The filtering approach. **Journal of fluid mechanics**, v. 238, p. 325 – 336, 1992.
- GIANNAKOPOULOS, G.; FROUZAKIS, C.; BOULOUCHOS, K.; FISCHER, P.; TOMBOULIDES, A. Direct numerical simulation of the flow in the intake pipe of an internal combustion engine. **International Journal of Heat and Fluid Flow**, v. 68, p. 257 – 268, 2017. ISSN 0142-727X.
- GOODWIN, D. G.; MOFFAT, H. K.; SPETH, R. L. **Cantera: an object-oriented software toolkit for chemical Kinetics, Thermodynamics, and Transport Processes**. 2017. Version 2.3.0. Available at: <<http://www.cantera.org>>. Access on: 30 Jun 2019.
- GORYNTSEV, D.; SADIKI, A.; KLEIN, M.; JANICKA, J. Analysis of cyclic variations of liquid fuel-air mixing processes in a realistic DISI IC-engine using Large Eddy Simulation. **International Journal of Heat and Fluid Flow**, v. 31, p. 845–849, 2010.
- GRAFTIEAUX, L.; MICHARD, M.; GROSJEAN, N. Combining PIV, POD and vortex identification algorithms for the study of unsteady turbulent swirling flows. **Measurement Science And Technology**, v. 12, p. 1422–1429, 2001.
- GUO, X.; LI, W.; IORIO, F. Convolutional neural networks for steady flow approximation. In: ACM SIGKDD CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING, 22, 2016, San Francisco. In: **SIGKDD Conference**. San Francisco: SIGKDD, 2016. p. 1–10.
- HAAGEN-SMIT, A. J. Chemistry and physiology of los angeles mmog. **Industrial and Engineering Chemistry**, v. 44, p. 1342 – 1346, 1952.
- HANLEY, S. **Clean technica: reducing carbon emissions — hybrid Vs. plug-in hybrid vs. battery electric**. 2019. Available at: <<https://cleantechnica.com/2019/06/14/reducing-carbon-emissions-hybrid-vs-plug-in-hybrid-vs-battery-electric/>>. Access on: 30 Jun 2019.
- HARLOW, F. H. Fluid dynamics in group t-3 los alamos national laboratory: (la-ur-03-3852). **Journal of Computational Physics**, v. 195, n. 2, p. 414 – 433, 2004. ISSN 0021-9991.

- HE, K.; ZHANG, X.; REN, S.; SUN, J. Deep residual learning for image recognition. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 29, 2016, Las Vegas, NV. In: **Proceedings of IEEE 29**. Las Vegas, NV: IEEE, 2016. p. 770–778.
- HEYWOOD, J. B. **Internal combustion engine fundamentals**. New York, USA: McGraw-Hill, 1988.
- HIERETH, H.; PRENNINGER, P. Introduction and short history of supercharging. In: HIERETH F.W. (Ed.). **Charging the Internal Combustion Engine. Powertrain**. Berlin: Springer, 2007. p. 1–4.
- HINCHET, R.; KHAN, U.; FALCONI, C.; KIM, S.-W. Piezoelectric properties in two-dimensional materials: simulations and experiments. **Materials Today**, 2018. ISSN 1369-7021.
- HONG, C.; TARNG, S. Direct measurement and computational analysis of turbulence length scales of a motored engine. **Experimental Thermal and Fluid Science**, v. 16, n. 4, p. 277 – 285, 1998. ISSN 0894-1777.
- HORROCKS, G. **A numerical study of a rotary valve internal combustion engine**. 2001. Thesis (PhD in Mechanical Engineering) - The University of Technology, Sydney, Australia. 463 p. Available at: <<https://opus.lib.uts.edu.au/bitstream/2100/248/15/02Whole.pdf>>. Access on: 30 Jun 2019.
- HOUDRY, E.; BURT, W. F.; PEW, A. E.; PETERS JR., W. A. The houdry process. **Oil and Gas Journal, Engineering and Operating**, v. 37, p. 40 – 45, 1938.
- HOUDRY, E. J. **Catalytic structure and composition**. US 2742437A, 24 may. 1949. Available at: <<https://patents.google.com/patent/US2742437>>. Access on: 30 Jun 2019.
- ICCT. **EU CO2 emission standards for passenger cars and light-commercial vehicles**. 2014. Available at: <[https://www.theicct.org/sites/default/files/publications/ICCTupdate\\_EU-95gram\\_jan2014.pdf](https://www.theicct.org/sites/default/files/publications/ICCTupdate_EU-95gram_jan2014.pdf)>. Access on: 30 Jun 2019.
- ILLGEN, S.; HÖCK, M. Electric vehicles in car sharing networks – challenges and simulation model analysis. **Transportation Research Part D: Transport and Environment**, v. 63, p. 377 – 387, 2018. ISSN 1361-9209.
- ISLAM, K. N.; HILDENBRAND, J.; HOSSAIN, M. M. Life cycle impacts of three-way ceramic honeycomb catalytic converter in terms of disability adjusted life year. **Journal of Cleaner Production**, v. 182, p. 600 – 615, 2018. ISSN 0959-6526.
- ISSA, R. Solution of the implicitly discretised fluid flow equations by operator-splitting. **Journal of Computational Physics**, v. 62, n. 1, p. 40 – 65, 1986. ISSN 0021-9991.
- ISSAWI, C. The 1973 oil crisis and after. **Journal of Post Keynesian Economics**, Routledge, v. 1, n. 2, p. 3–26, 1978.
- JAIN, S. K.; AGGARWAL, S. K. Compositional effects on the ignition and combustion of low octane fuels under diesel conditions. **Fuel**, v. 220, p. 654 – 670, 2018. ISSN 0016-2361.
- JANAS, P. **Large eddy simulation of in-cylinder phenomena in spark ignition engines**. 2017. Thesis (PhD in Mechanical Engineering) - Universität Duisburg-Essen, Duisburg, Germany. 152 p. Available at: <[https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico\\_derivate\\_00043957/Diss\\_PJanas.pdf](https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico_derivate_00043957/Diss_PJanas.pdf)>. Access on: 30 Jun 2019.
- JANAS, P.; RIBEIRO, M.; KEMPF, A.; SCHILD, M.; KAISER, S. Penetration of the flame into the top-land crevice - Large-Eddy Simulation and Experimental High-Speed Visualization. **SAE Technical Paper**, n. 2015-01-1907, 2015.

- JASAK, H.; TUKOVIĆ, Z. Automatic mesh motion for the unstructured finite volume method. **Transactions of FAMENA**, v. 30, n. 2, p. 1–20, 2006.
- JONES, W.; LAUNDER, B. The prediction of laminarization with a two-equation model of turbulence. **International Journal of Heat and Mass Transfer**, v. 15, n. 2, p. 301 – 314, 1972. ISSN 0017-9310.
- KAWAMOTO, N.; NAIKI, K.; KAWAI, T.; SHIKIDA, T.; TOMATSURI, M. Development of new 1.8-liter engine for hybrid vehicles. In: SAE INTERNATIONAL CONGRESS AND EXPOSITION, 143, 2009, Detroit. In: **SAE International**. Detroit: SAE, 2009. p. 1–9.
- KIM, W.-W.; MENON, S. A new dynamic one-equation subgrid-scale model for large eddy simulations. In: AMERICAN INSTITUTE OF AERONAUTICS AND ASTRONAUTICS. **33rd Aerospace Sciences Meeting and Exhibit**. Reno, NV, 1995. AIAA 95-0356, p. 9.
- KINGMA, D. P.; BA, J. **Adam: A method for stochastic optimization**. 2017. 1412.6980v9 [cs.LG]. Available at: <<https://arxiv.org/abs/1412.6980>> Access on: 30 Jul 2019.
- KOLMOGOROV, A. N. The local structure of turbulence in incompressible viscous fluids at very large reynolds numbers. **Dokl. Akad. Nauk. SSSR**, v. 30, p. 299 – 303, 1941.
- KRISHNA, A. S.; MALLIKARJUNA, J.; KUMAR, D. Effect of engine parameters on in-cylinder flows in a two-stroke gasoline direct injection engine. **Applied Energy**, v. 176, p. 282 – 294, 2016. ISSN 0306-2619.
- KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. In: INTERNATIONAL CONFERENCE ON NEURAL INFORMATION PROCESSING SYSTEMS, 25, 2012, Lake Tahoe, NV. In: **Proceedings of the NIPS 12**. Lake Tahoe, NV: Curran Associates Inc., 2016. p. 1097–1105.
- LABERGUE, A.; DEPRÉDURAND; DELCONTE, A.; CASTANET, G.; LEMOINE, F. New insight into two-color lif thermometry applied to temperature measurements of droplets. **Experiments in Fluids**, v. 49, n. 2, p. 547–556, 2010.
- LECUN, Y.; BOTTOU, L.; BENGIO, Y.; HAFFNER, P. Gradient-based learning applied to document recognition. In: IEEE INTERNATIONAL CONFERENCE, 86, 1998, San Diego, CA. In: **Proceedings of the IEEE**. San Diego, CA: IEEE, 1998. p. 2278–2324.
- LIBERMAN, M. A. Internal combustion engines. In: LIBERMAN, M. A. (Ed.). **Introduction to physics and chemistry of combustion**. Berlin: Springer, 2008. p. 53–72.
- LIN, M.; CHEN, Q.; YAN, S. **Network in network**. 2014. 1312.4400v3 [cs.NE]. Available at: <<https://arxiv.org/abs/1312.4400>> Access on: 30 Jul 2019.
- LINNAINMAA, S. **The representation of the cumulative rounding error of an algorithm as a taylor expansion of the local rounding errors**. 1970. Dissertation (Masters in Computer Sciences) - University of Helsinki, Helsinki, Finland. 125 p.
- LUCCHINI, T.; D'ERRICO, G.; ETTORRE, D. Numerical investigation of the spray–mesh–turbulence interactions for high-pressure, evaporating sprays at engine conditions. **International Journal of Heat and Fluid Flow**, v. 32, n. 1, p. 285 – 297, 2011. ISSN 0142-727X.
- MANDANIS, C.; SCHMITT, M.; KOCH, J.; WRIGHT, Y. M.; BOULOUCOS, K. Wall heat flux and thermal stratification investigations during the compression stroke of an engine-like geometry: a comparison between les and dns. **Flow, Turbulence and Combustion**, v. 100, n. 3, p. 769–795, Apr 2018.

- MAROTEAUX, F. Development of a two-part n-heptane oxidation mechanism for two stage combustion process in internal combustion engines. **Combustion and Flame**, v. 186, p. 1 – 16, 2017. ISSN 0010-2180.
- MARR, M. A.; WALLACE, J. S.; CHANDRA, S.; PERSHIN, L.; MOSTAGHIMI, J. A fast response thermocouple for internal combustion engine surface temperature measurements. **Experimental thermal and fluid science**, v. 34, n. 2, p. 183 – 189, 2010. ISSN 0894-1777.
- MENTER, F. R. Two-equation eddy-viscosity turbulence models for engineering applications. **AIAA Journal**, v. 62, n. 8, p. 1598–1605, 1994.
- MERKER, G.; SCHWARZ, C.; STIESCH; OTTO, F. **Simulating combustion**. Berlin, Germany: Springer Verlag, 2006.
- MGHPCC. **The Massachusetts green high performance cluster**. 2019. Available at: <<https://www.mghpcc.org/>>. Access on: 30 Jun 2019.
- NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING, 27, 2010, Haifa. In: **Proceedings of the 27 ICML**. Haifa: Omnipress, 2010. p. 807–814.
- NG, A. **Machine learning yearning: technical strategies for AI engineers in the era of deep learning**. Palo Alto, CA, United States: Online Draft, 2017.
- NGUYEN, T. **Large eddy simulation of internal combustion engine with efficient, particle-based moving immersed boundaries**. 2017. Thesis (PhD in Mechanical Engineering) - Universität Duisburg-Essen, Duisburg, Germany. 213 p. Available at: <[https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico\\_derivate\\_00044045/Diss\\_Nguyen.pdf](https://duepublico2.uni-due.de/servlets/MCRFileNodeServlet/duepublico_derivate_00044045/Diss_Nguyen.pdf)>. Access on: 30 Jun 2019.
- NOBLE, W. M. Air pollution effects, pattern of damage produced on vegetation by smog. **Journal of Agriculture and Food Chemistry**, v. 3, p. 330 – 332, 1955.
- OLBRICHT, C.; STEIN, O.; JANICKA, J.; OIJEN, J. van; WYSOCKI, S.; KEMPF, A. Les of lifted flames in a gas turbine model combustor using top-hat filtered pfgm chemistry. **Fuel**, v. 96, p. 100 – 107, 2012. ISSN 0016-2361.
- OPENFOAM. **User guide version 5.0**. 2017. Available at: <<http://openfoam.org>>. Access on: 30 Jun 2019.
- O’ROURKE, P. J. **Collective drop effects on vaporizing liquid sprays**. 1981. Thesis (PhD in Mechanical Engineering) - Princeton University, New Jersey, USA. 195 p. Available at: <<https://www.osti.gov/biblio/5201366-collective-drop-effects-vaporizing-liquid-sprays>>. Access on: 30 Jun 2019.
- PARRISH, S. E. **Drop size measurements: engine combustion network 3rd workshop**. 2014. Available at: <<https://ecn.sandia.gov/ecn-workshop/ecn3-proceedings/>>. Access on: 30 Jun 2019.
- PARRISH, S. E. Evaluation of liquid and vapor penetration of sprays from a multi-hole gasoline fuel injector operating under engine-like conditions. **SAE Int. J. Engines**, SAE International, v. 7, p. 1017–1033, 04 2014.
- PASTOR, J.; GARCÍA-OLIVER, J.; LÓPEZ, J.; VERA-TUDELA, W. An experimental study of the effects of fuel properties on reactive spray evolution using primary reference fuels. **Fuel**, v. 163, p. 260 – 270, 2016. ISSN 0016-2361.

- PATANKAR, S.; SPALDING, D. A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows. **International Journal of Heat and Mass Transfer**, v. 15, n. 10, p. 1787 – 1806, 1972. ISSN 0017-9310.
- PATANKAR, S. V. **Numerical heat transfer and fluid flow**. Washington: Hemisphere Publishing Corporation, 1980.
- PETERSON, B.; BAUM, E.; BÖHM, B.; SICK, V.; DREIZLER, A. High-speed piv and lif imaging of temperature stratification in an internal combustion engine. **Proceedings of the Combustion Institute**, v. 34, n. 2, p. 3653 – 3660, 2013. ISSN 1540-7489.
- PETERSON, B.; BAUM, E.; DING, C. P.; MICHAELIS, D.; DREIZLER, A.; BÖHM, B. Evaluation of spray-induced turbulence during the induction stroke of a four-stroke single-cylinder optical engine. In: INTERNATIONAL SYMPOSIUM ON THE APPLICATION OF LASER AND IMAGING TECHNIQUES TO FLUID MECHANICS, 18, 2016, Lisbon. In: **Proceedings of the Lisbon Symposium**. Lisbon: Lisbon symposium, 2016. p. 1–19.
- PETERSON, B.; BAUM, E.; DING, C.-P.; MICHAELIS, D.; DREIZLER, A.; BÖHM, B. Assessment and application of tomographic piv for the spray-induced flow in an ic engine. **Proceedings of the Combustion Institute**, v. 36, n. 3, p. 3467 – 3475, 2017. ISSN 1540-7489.
- PETERSON, B.; BAUM, E.; DING, C.-P.; MICHAELIS, D.; DREIZLER, A.; BÖHM, B. Assessment and application of tomographic piv for the spray-induced flow in an ic engine. **Proceedings of the Combustion Institute**, v. 36, n. 3, p. 3467 – 3475, 2017. ISSN 1540-7489.
- PETERSON, B.; REUSS, D. L.; SICK, V. On the ignition and flame development in a spray-guided direct-injection spark-ignition engine. **Combustion and Flame**, v. 161, p. 240–255, 2014.
- PICKETT, L. M. Low flame temperature limits for mixing-controlled diesel combustion. **Proceedings of the Combustion Institute**, v. 30, n. 2, p. 2727 – 2735, 2005. ISSN 1540-7489.
- POPE, S. B. Ten questions concerning the large-eddy simulation of turbulent flows. **New Journal of Physics**, v. 6, n. 1, p. 35, 2004.
- POPE, S. B. Simple models of turbulent flows. **Physics of Fluids**, v. 23, p. 011301:1–20, 2011.
- PORAN, A.; THAWKO, A.; EYAL, A.; TARTAKOVSKY, L. Direct injection internal combustion engine with high-pressure thermochemical recuperation – experimental study of the first prototype. **International Journal of Hydrogen Energy**, v. 43, n. 27, p. 11969 – 11980, 2018. ISSN 0360-3199.
- R. J. Kee and F. M. Rupley and J. A. Miller. **CHEMKIN Reaction design: San Diego**. 2013. CHEMKIN 10131. Available at: <<http://www.reactiondesign.com/>>. Access on: 30 Jun 2019.
- RABAULT, J.; VERNET, J. A.; LINDGREN, B.; ALFREDSSON, P. H. A study using piv of the intake flow in a diesel engine cylinder. **International Journal of Heat and Fluid Flow**, v. 62, p. 56 – 67, 2016. ISSN 0142-727X.
- RAFFEL, M.; WILLERT, C. E.; WERELEY, S. T.; KOMPENHANS, K. **Particle image velocimetry**. Berlin: Springer, 2007.
- RAISSI, M.; PERDIKARIS, P.; KARNIADAKIS, G. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. **Journal of Computational Physics**, v. 378, p. 686 – 707, 2019. ISSN 0021-9991.

- RAISSI, M.; WANG, Z.; TRIANTAFYLLOU, M. S.; KARNIADAKIS, G. E. **Deep learning of vortex induced vibrations**. 2018. ArXiv 1808.08952 [physics.flu-dyn]. Available at: <<https://arxiv.org/abs/1808.08952>> Access on: 30 Jul 2019.
- RANZ, W. E.; MARSHALL, W. R. Evaporation from drops. **Chemical Engineering Progress**, v. 48, n. 3, p. 141–146, 1952.
- REITZ, R. D. Modeling atomization processes in high-pressure vaporizing sprays. **Atomisation and Spray Technology**, v. 3, p. 309–337, 1987.
- REITZ, R. D.; BRACCO, F. V. **The encyclopedia of fluid mechanics**. [S.l.]: Gulf Publishing, 1986.
- REITZ, R. D.; DIWAKAR, R. Structure of high-pressure fuel sprays. **SAE Technical Paper**, n. 870598, 1987.
- RIBEIRO, M. D. **Engine LES with fuel-spray modeling**. 2018. Dissertação (Mestrado em engenharia Mecânica) - Faculdade de Engenharia do Campus de Guaratinguetá, Universidade Estadual Paulista, Guaratinguetá. 118p. Available at: <<http://repositorio.unesp.br/handle/11449/127949>>. Access on: 30 Jun 2019.
- RICHARD, S.; COLIN, O.; VERMOREL, O.; BENKENIDA, A.; ANGELBERGER, C.; VEYNANTE, D. Towards large eddy simulation of combustion in spark ignition engines. **Proceedings of the Combustion Institute**, v. 31, p. 3059–3066, 2007.
- RITTLER, A.; PROCH, F.; KEMPF, A. M. Les of the sydney piloted spray flame series with the pfgm/atf approach and different sub-filter models. **Combustion and flame**, v. 162, n. 4, p. 1575 – 1598, 2015. ISSN 0010-2180.
- SASS, F. Karl benz in mannheim (1876–1890). In: SASS F. (Ed.). **Geschichte des deutschen verbrennungsmotorenbaues**. Berlin: Springer, 1962. p. 1–4.
- SCARANO, F. Tomographic piv: principles and practice. **Measurement Science and Technology**, v. 24, n. 1, p. 012001, 2013.
- SCHMIDT, D. P.; RUTLAND, C. A new droplet collision algorithm. **Journal of Computational Physics**, v. 164, n. 1, p. 62 – 80, 2000. ISSN 0021-9991.
- SCHMITT, M. **Direct numerical simulations in engine-like geometries**. 2014. Thesis (PhD in Mechanical Engineering) - ETH Zurich, Zurich, Switzerland. 227 p. Available at: <<http://e-collection.library.ethz.ch/eserv/eth:47222/eth-47222-02.pdf>>. Access on: 30 Jun 2019.
- SCHOETTLIN, C. E.; LANDAU, E. Air pollution and asthmatic attacks in the los angeles area. **Public Health Reports**, v. 76, p. 545 – 551, 1961.
- SCHULZ, C. Optical diagnostics in diesel combustion engines. In: ZHAO H. (Ed.). **Advanced direct injection combustion engine technologies and development**. Cambridge: Woodhead Publishing, 2010. p. 617–643.
- SDSC. **Comet user guide**. 2019. Available at: <[https://www.sdsc.edu/support/user\\_guides/comet.html](https://www.sdsc.edu/support/user_guides/comet.html)>. Access on: 30 Jun 2019.
- SIMONYAN, K.; ZISSERMAN, A. **Very deep convolutional networks for large-scale image recognition**. 2015. 1409.1556v6 [cs.CV]. Available at: <<https://arxiv.org/abs/1409.1556>> Access on: 30 Jul 2019.



SJÖBERG, M.; REUSS, D. L. High-speed imaging of spray-guided disi engine combustion with near-tdc injection of e85 for ultra-low no and soot. **Proceedings of the Combustion Institute**, v. 34, n. 2, p. 2933 – 2940, 2013. ISSN 1540-7489.

SKEEN, S. A.; MANIN, J.; PICKETT, L. M. Simultaneous formaldehyde plif and high-speed schlieren imaging for ignition visualization in high-pressure spray flames. **Proceedings of the Combustion Institute**, v. 35, n. 3, p. 3167 – 3174, 2015. ISSN 1540-7489.

SMAGORINSKY, J. General circulation experiments with the primitive equations. **Monthly Weather Review**, v. 91, n. 3, p. 99–164, 1963.

SMITH, A.; DAVIES, H. A review of the history of emission legislation, urban and national transport trends and their impact on transport emissions. **Transactions on the Built Environment**, v. 23, p. 293 – 302, 1996.

SPHICAS, P.; PICKETT, L. M.; SKEEN, S.; FRANK, J.; LUCCHINI, T.; SINOIR, D.; D'ERRICO, G.; SAHA, K.; SOM, S. A comparison of experimental and modeled velocity in gasoline direct-injection sprays with plume interaction and collapse. **SAE International Journal of Fuels and Lubricants**, v. 10, n. 1, p. 184–201, mar 2017.

SPHICAS, P.; PICKETT, L. M.; SKEEN, S. A.; FRANK, J. H. Inter-plume aerodynamics for gasoline spray collapse. **International Journal of Engine Research**, v. 0, n. 0, p. 1–20, 2017.

SZEGEDY, C.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCKE, V.; RABINOVICH, A. Going deeper with convolutions. In: IEEE CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION (CVPR), 28, 2015, Boston, MA. In: **Proceedings of the IEEE 28**. Boston, MA: IEEE, 2015. p. 1063–6919.

TENSORFLOW. **Get started with tensorflow**. 2019. Available at: <<https://www.tensorflow.org/tutorials>>. Access on: 30 Jun 2019.

TOLLMIEN, W.; SCHLICHTING, H.; GÖRTLER, H.; RIEGELS, F. W. Über die ausgebildete turbulenz. In: RIEGELS F.W. (Ed.). **Ludwig Prandtl gesammelte abhandlungen**. Berlin: Springer, 1961. p. 62–77.

TOMPSON, J.; SCHLACHTER, K.; SPRECHMANN, P.; PERLIN, K. **Accelerating eulerian fluid simulation with convolutional networks**. 2017. ArXiv. 1607.03597v6 [cs.CV]. Available at: <<https://arxiv.org/abs/1607.03597>> Access on: 30 Jul 2019.

TORMOS, B.; MARTÍN, J.; CARREÑO, R.; RAMÍREZ, L. A general model to evaluate mechanical losses and auxiliary energy consumption in reciprocating internal combustion engines. **Tribology International**, v. 123, p. 161 – 179, 2018. ISSN 0301-679X.

TU Darmstadt. **Darmstadt engine workshop**. 2018. Available at: <[https://www.rsm.tu-darmstadt.de/home\\_rsm/news\\_rsm/darmstadt\\_engine\\_workshop.de.jsp](https://www.rsm.tu-darmstadt.de/home_rsm/news_rsm/darmstadt_engine_workshop.de.jsp)>. Access on: 30 Jun 2019.

UMICH. **TCC engine details**. 2018. Available at: <<https://sites.google.com/a/umich.edu/volker-sick/Research/les-working-group/tcc-engine-details>>. Access on: 30 Jun 2019.

VERMOREL, O.; RICHARD, S.; COLIN, O.; ANGELBERGER, C.; BENKENIDA, A.; VEYNANTE, D. Towards the understanding of cyclic variability in a spark ignited engine using multi-cycle les. **Combustion and Flame**, v. 156, n. 8, p. 1525 – 1541, 2009. ISSN 0010-2180.

VERSTEEG, H. K.; MALALASEKERA, W. **An introduction to computational fluid dynamics: the finite volume method**. Harlow, England: Pearson, 2007.

VUORINEN, V. A.; HILLAMO, H.; KAARIO, O.; NUUTINEN, M.; LARMI, M.; FUCHS, L. Large eddy simulation of droplets stokes number effects on mixture quality in fuel sprays. **Atomization and Sprays**, v. 20, p. 435–451, 2010.

VUORINEN, V. A.; HILLAMO, H.; KAARIO, O.; LARMI, M.; FUCHS, L. Large eddy simulation of droplets stokes number effects on turbulent spray shape. **Atomization and Sprays**, v. 20, p. 93–114, 2010.

VUORINEN, V. A.; HILLAMO, H.; KAARIO, O.; NUUTINEN, M.; LARMI, M.; FUCHS, L. Effect of droplet size and atomization on spray formation: a priori study using large-eddy simulation. **Flow Turbulence Combust**, v. 86, p. 533–561, 2011.

WANG, H.; YAO, M.; REITZ, R. D. Development of a reduced primary reference fuel mechanism for internal combustion engine simulations. **Energy & Fuels**, v. 27, n. 12, p. 7843–7853, 2013.

WARNATZ, J.; MAAS, U.; DIBBLE, R. W. **Combustion: physical and chemical fundamentals, modeling and simulation, experiments, pollutant formation**. Berlin, Germany: Springer Verlag, 2006.

WELLER, H. G.; TABOR, G.; JASAK, H.; FUREBY, C. A tensorial approach to computational continuum mechanics using object-oriented techniques. **Computer in Physics**, v. 12, n. 6, p. 620–631, 1998.

WESTBROOK, C.; PITZ, W.; MEHL, M.; CURRAN, H. Detailed chemical kinetic reaction mechanisms for primary reference fuels for diesel cetane number and spark-ignition octane number. **Proceedings of the Combustion Institute**, v. 33, n. 1, p. 185 – 192, 2011. ISSN 1540-7489.

WESTBROOK, C. K.; DRYER, F. L. Simplified reaction mechanisms for the oxidation of hydrocarbon fuels in flames. **Combustion Science and Technology**, Taylor and Francis, v. 27, n. 1-2, p. 31–43, 1981.

WESTBROOK, C. K.; MIZOBUCHI, Y.; POINSOT, T. J.; SMITH, P. J.; WARNATZ, J. Computational combustion. **Proceedings of the Combustion Institute**, v. 30, n. 1, p. 125 – 157, 2005. ISSN 1540-7489.

WIERZBA, A. Deformation and breakup of liquid drops in a gas stream at nearly critical Weber numbers. **Experiments in Fluids**, v. 9, p. 59–64, 1990.

WIKIMEDIA. **4 stroke cycle**. 2018. Available at: <[https://commons.wikimedia.org/wiki/File:Four\\_stroke\\_cycle\\_start.png](https://commons.wikimedia.org/wiki/File:Four_stroke_cycle_start.png)>. Access on: 30 Jun 2019.

WIKIMEDIA. **Boundary layer**. 2018. Available at: <[https://en.wikipedia.org/wiki/Boundary\\_layer#/media/File:Boundarylayer.png](https://en.wikipedia.org/wiki/Boundary_layer#/media/File:Boundarylayer.png)>. Access on: 30 Jun 2019.

WIKIMEDIA. **Cutaway view of a V-6 engine**. 2018. Available at: <[https://commons.wikimedia.org/wiki/File:IC\\_engine.JPG](https://commons.wikimedia.org/wiki/File:IC_engine.JPG)> Access on: 30 Jul 2019.

WIKIMEDIA. **Energy spectrum of turbulent cascade**. 2018. Available at: <<https://commons.wikimedia.org/wiki/File:Schematic-illustration-of-the-energy-spectrum-of-turbulent-velocity-cascade.png>>. Access on: 30 Jun 2019.

WIKIMEDIA. **Neural Net Graph**. 2018. Available at: <<https://openclipart.org/detail/290666/neural-net-graph>>. Access on: 30 Jun 2019.

WIKIMEDIA. **One-dimensional volume**. 2018. Available at: <[https://commons.wikimedia.org/wiki/File:Discretisation\\_for\\_Control\\_volume\\_in\\_TVD\\_scheme.png](https://commons.wikimedia.org/wiki/File:Discretisation_for_Control_volume_in_TVD_scheme.png)>. Access on: 30 Jun 2019.

WIKIMEDIA. **Pressure volume element**. 2018. Available at: <[https://upload.wikimedia.org/wikipedia/commons/c/c1/Pressure\\_on\\_an\\_infinitesimal\\_volume\\_element.svg](https://upload.wikimedia.org/wikipedia/commons/c/c1/Pressure_on_an_infinitesimal_volume_element.svg)>. Access on: 30 Jun 2019.

WIKIMEDIA. **Shear stress infinitesimal volume element**. 2018. Available at: <[https://commons.wikimedia.org/wiki/File:Shear\\_stress\\_infinitesimal\\_volume\\_element.svg](https://commons.wikimedia.org/wiki/File:Shear_stress_infinitesimal_volume_element.svg)>. Access on: 30 Jun 2019.

WIKIMEDIA. **Unit injector**. 2018. Available at: <<https://commons.wikimedia.org/wiki/File:Injector3.gif>>. Access on: 30 Jun 2019.

WILCOX, D. C. Re-assessment of the scale-determining equation for advanced turbulence models. **AIAA Journal**, v. 26, n. 11, p. 1299–1310, 1988.

ZHOU, D.; YANG, W.; LI, J.; TAY, K. L. Simplified fuel cracking process in reduced mechanism development: Prf – pah kinetic models for combustion and soot prediction. **Fuel**, v. 182, p. 831 – 841, 2016. ISSN 0016-2361.

ZUO, B.; GOMES, A. M.; RUTLAND, C. J. Modelling superheated fuel sprays and vaporization. **Int J Engine Research**, v. 1, p. 321–336, 2000.

## APPENDIX A – GOVERNING EQUATIONS

### A.1 EQUATIONS DERIVATION

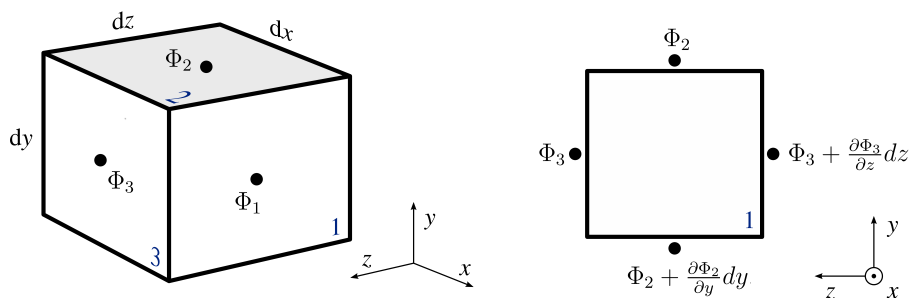
The equations presented in chapter 3, section 1, are obtained by balancing mass, momentum, energy, and species flowing through the boundary surfaces of each domain volume element at every simulation time step. These equations have the general form shown in (36):

$$\frac{\partial}{\partial t}(\rho\Phi) + \frac{\partial}{\partial x_j}(\rho u_j \Phi) = \frac{\partial}{\partial x_j} \left( \Gamma_\Phi \frac{\partial \Phi}{\partial x_j} \right) + S_\Phi \quad (36)$$

The first term on the left hand side of the above equation, which takes into account the changes of a given property  $\Phi$  in time, is known as the accumulation term. The next term describes the convective transport of  $\Phi$  throughout the domain, being thus named the convective term. Furthermore, the first term on the right hand side represents the diffusive transport of  $\Phi$ , being  $\Gamma_\Phi$  known as the general diffusion coefficient. Finally, the last term expresses the presence of a source or sink of  $\Phi$ .

The derivation process of equations (1) through (3), also known as the Navier-Stokes equations, will be now here depicted. First, one needs to have a clear definition of the volume element where the conservation equations are applied. In three-dimensional CFD simulations of practical interest for engineering, the computational domain (also known as mesh or grid) may contain millions of these cell elements in a variety of shapes. For the sake of simplicity, here the control volume will be considered a hexahedron, as show in Figure 77.

Figure 77 – Control volume



Source: Adapted from "pressure volume element" (licensed under Creative Commons) by Wikimedia (2018g).

In usual CFD codes, every particular flow property  $\Phi$  is calculated and stored in the center of the volume element. However, as the finite volume method requires the evaluation of inflowing and outflowing fluxes through the element boundaries in order to assure conservation, the value of  $\Phi$  must also be interpolated to and stored at the boundary surfaces of each element. The gradient of  $\Phi$  in a certain direction (say along the  $z$  axis on plane 1 of Figure 77) can be obtained by means of a truncated Taylor series, e.g.  $\Phi_3 + \frac{\partial \Phi_3}{\partial z} dz$ .

With use of the above definitions, the first equation to be derived is the mass conservation or

continuity equation. At one particular time, the rate of accumulation of mass in a cell is shown in (37).

$$\dot{M}_1 = \frac{\partial \rho}{\partial t} . dx dy dz \quad (37)$$

The inflowing mass in the cell along the x axis (through area dydz) is shown in equation (38), whereas the outflowing mass through the same area is shown in equation (39).

$$\dot{M}_2 = \rho u . dy dz \quad (38)$$

$$\dot{M}_3 = \left( \rho u + \frac{\partial \rho u}{\partial x} dx \right) . dy dz \quad (39)$$

The balance between (38) and (39) results in equation (40). The same should be done for fluxes in areas dx dy and dx dz.

$$\dot{M}_4 = - \frac{\partial \rho u}{\partial x} . dx dy dz \quad (40)$$

The change in mass in a cell at one particular time step should be equal to the balance of mass flowing through its boundary surfaces, resulting in equation (41).

$$\frac{\partial \rho}{\partial t} . dx dy dz + \left( \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} \right) . dx dy dz = 0 \quad (41)$$

The division of the previous equation by the element volume dV (dx dy dz) results in a more common form shown in equation (42).

$$\frac{\partial \rho}{\partial t} + \left( \frac{\partial \rho u}{\partial x} + \frac{\partial \rho v}{\partial y} + \frac{\partial \rho w}{\partial z} \right) = 0 \quad (42)$$

which can be described more compactly with the use of the index notation, as shown in equation (43). In this notation, a subscript represents each of the spacial components.

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \quad (43)$$

In a similar way, momentum must also be conserved. For this, the accumulation of momentum due to the velocity components in each direction (u in x, v in y, and w in z) is evaluated. For the u velocity component, this is done as shown in equation (44).

$$\dot{F}_1 = \frac{\partial \rho u}{\partial t} . dx dy dz \quad (44)$$

Momentum is also transported throughout the computational domain by the action of convection. The momentum fluxes of velocity component u in directions x, y, and z are shown in equations (45), (46), and (47), respectively.

$$\dot{F}_2 = \left( \left( (\rho u) u + \frac{\partial (\rho u) u}{\partial x} dx \right) - (\rho u) u \right) . dy dz \quad (45)$$

$$\dot{F}_3 = \left( \left( (\rho u)v + \frac{\partial(\rho u)v}{\partial y} dy \right) - (\rho u)v \right) . dx dz \quad (46)$$

$$\dot{F}_4 = \left( \left( (\rho u)w + \frac{\partial(\rho u)w}{\partial y} dz \right) - (\rho u)w \right) . dx dy \quad (47)$$

which can be described in a single term as shown in equation (48):

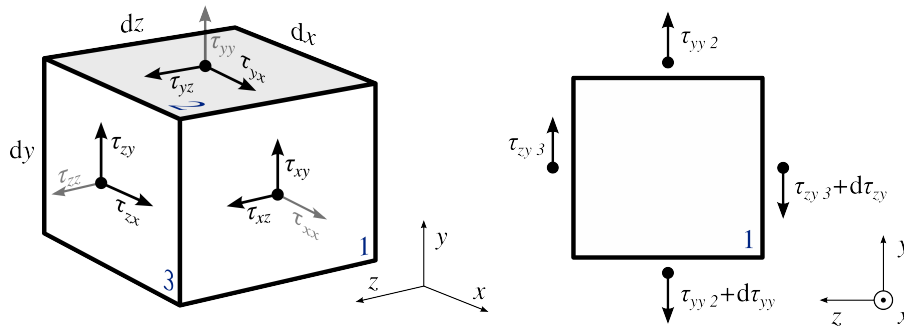
$$\dot{F}_5 = \left( \frac{\partial \rho u v}{\partial x} + \frac{\partial \rho u w}{\partial y} + \frac{\partial \rho u w}{\partial z} \right) . dx dy dz \quad (48)$$

Moreover, the forces acting on the volume element, such as pressure, viscous, and gravitational forces, have to be taken into account. Pressure forces caused by the flow act over the surface areas of the control volume for all dimensions considered. The balance of pressure force in x is shown in (49):

$$\dot{F}_6 = \left( p - \left( p + \frac{\partial p}{\partial x} \right) \right) . dy dz \quad (49)$$

Furthermore, the diffusive transport caused by molecular transfer is represented by the balance of all components of the stress tensor  $\tau$ , that can be pointed normally (normal stress) or tangentially (shear stress) to the surface area (see Figure 78). For each direction, normal and shear stresses are balanced as shown in equations (50), (51), and (52).

Figure 78 – Shear forces on volume element



Source: Adapted from "Shear stress infinitesimal volume element" (licensed under Creative Commons) by Wikimedia (2018h).

$$\dot{F}_7 = \left( \tau_{xx} - \left( \tau_{xx} + \frac{\partial \tau_{xx}}{\partial x} dx \right) \right) . dy dz \quad (50)$$

$$\dot{F}_8 = \left( \tau_{yx} - \left( \tau_{yx} + \frac{\partial \tau_{yx}}{\partial y} dy \right) \right) . dx dz \quad (51)$$

$$\dot{F}_9 = \left( \tau_{zx} - \left( \tau_{zx} + \frac{\partial \tau_{zx}}{\partial z} dz \right) \right) . dx dy \quad (52)$$

which, in the case of a Newtonian fluid, can be modelled as in equations (53), (54), and (55):

$$\tau_{xx} = -2\mu \frac{\partial u}{\partial x} + \frac{2}{3}\mu \left( \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} \right) \quad (53)$$

$$\tau_{xy} = -\mu \left( \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) \quad (54)$$

$$\tau_{xz} = -\mu \left( \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \right) \quad (55)$$

Finally, the effect of gravity is given in equation (56):

$$F_{10} = \rho g_x \cdot dx dy dz \quad (56)$$

In order for momentum to be conserved, the accumulation of momentum (44) plus the convective transport of momentum (48) must equal the summation of all forces experienced by the control volume, i.e., pressure (49), viscous (50 through 52), and gravitational forces (56), in all directions. The final form for the conservation of momentum in index notation is presented in equation (57):

$$\frac{\partial \rho u_i}{\partial t} + \frac{\rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} - \frac{\partial \tau_{ji}}{\partial x_j} + \rho g_i \quad (57)$$

For deriving the energy equation, it is usually assumed that the total energy accumulation in a volume element must be equal to the balance of inflowing/outflowing energy plus the work expended on the fluid. The total energy can be expressed either in terms of inner energy (58) or enthalpy (59), with the latter being adopted here.

$$e_t = e + \frac{1}{2}(u^2 + v^2 + w^2) = e + K \quad (58)$$

$$h_t = e_t + \frac{p}{\rho} \quad (59)$$

As previously done for the conservation of mass and momentum, the first step is the definition of the total energy accumulation term in equation (60):

$$\dot{E}_1 = \frac{\partial}{\partial t} \left[ \rho \left( h_t - \frac{p}{\rho} \right) \right] \cdot dx dy dz \quad (60)$$

One more time the energy flux through the element surfaces must be evaluated for all types of energy, i.e. the convective, and diffusive transport of enthalpy (thermal conductivity), as show in equations (61) and (62), respectively. The effect of radiation transport is disregarded in this work.

$$\dot{E}_2 = -\frac{\partial}{\partial x_i} \left[ \rho u_i \left( h_t - \frac{p}{\rho} \right) \right] \cdot dx dy dz \quad (61)$$

$$\dot{E}_3 = -\frac{\partial}{\partial x_i} (q_{cond}) \cdot dx dy dz \quad (62)$$

The thermal conductivity  $q_{cond}$  is modelled by the Fourier law for  $h_t$  ( $q_{cond} = -\nabla \alpha_{eff} h_t$ ), in which  $\alpha_{eff}$  represents the sum of laminar and turbulent thermal diffusivity. Moreover, work is expended on

the fluid by the actions of molecular, pressure, and mass forces as given in (63), (64), and (65):

$$\dot{E}_4 = -\frac{\partial}{\partial x_i}(\tau_{ij}u_j) \quad (63)$$

$$\dot{E}_5 = -\frac{\partial}{\partial x_i}(u_i p) \quad (64)$$

$$\dot{E}_6 = \rho g_i \cdot u_i \quad (65)$$

Equalizing the accumulation term (60) with the summation of further energy terms (61-65) results in the energy conservation equation, shown in index notation in equation (66):

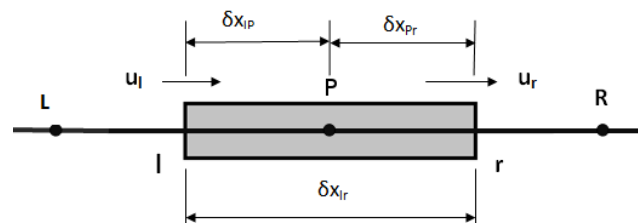
$$\frac{\partial(\rho h_t)}{\partial t} + \frac{\partial(\rho h_t u_i)}{\partial x_i} = -\frac{\partial}{\partial x_i}(q_{cond}) - \frac{\partial}{\partial x_i}(\tau_{ij}u_j) + \frac{\partial p}{\partial t} + \rho g_i \cdot u_i \quad (66)$$

The same procedure described above can be used to determine any another conservation law within the numerical framework, such as the species conservation in equation (4).

## A.2 NUMERICAL SOLUTION

Now that the governing equations have been derived, numerical methods for their solution can be provided. In order to come up with a solution procedure, it is first necessary to discretize the equations presented above in a framework suitable for the application of the finite volume method commonly used in CFD calculations. This requires to rewrite the partial derivative terms of the set of non-linear partial differential equations in terms of finite differences, resulting in a system of algebraic equations that can be linearized and resolved by a linear matrix solver. Therefore, this section will focus on describing such discretization methods of the fluid governing equations, as well as the fundamental principals of CFD solution algorithms, more specifically, the SIMPLE method proposed by Patankar (1980) and the PISO method proposed by Issa (1986).

Figure 79 – One-dimensional control volume



Source: Wikimedia (2018f).

First, all terms of equation (36), i.e. accumulation, convection, diffusion and source/sink, must be discretized. Assuming the velocity field is known, the discretized forms of convection and diffusion fluxes of  $\Phi$  along the one-dimensional control volume in Figure 79 are shown in equation (67) and



equation (68), respectively:

$$\left(\Gamma A \frac{\partial \Phi}{\partial x}\right)_r - \left(\Gamma A \frac{\partial \Phi}{\partial x}\right)_l = \frac{\Gamma_r}{\delta x_{PR}} A_r (\Phi_R - \Phi_P) - \frac{\Gamma_l}{\delta x_{PL}} A_l (\Phi_P - \Phi_L) \quad (67)$$

$$(\rho u A \Phi)_r - (\rho u A \Phi)_l = \rho_r u_r A_r \Phi_r - \rho_l u_l A_w \Phi_l \quad (68)$$

for which the definition of variables  $F$  and  $D$  to represent the convective mass flux and the mass diffusion per unit area is convenient, as show in equation (69):

$$F = \rho u \quad \text{and} \quad D = \frac{\Gamma}{\partial x} \quad (69)$$

thus, equations (67) and (68) can be rewritten in a per unit area basis, assuming that  $A_l = A_r = A$ , as follows in (70) and (71):

$$D_r (\Phi_R - \Phi_P) - D_l (\Phi_P - \Phi_L) \quad (70)$$

$$F_r \Phi_r - F_l \Phi_l \quad (71)$$

It should be noted that some quantities must be evaluated at cell centers (represented by capital letters L at the left, R at the right, and P at the current cell), whereas others are evaluated at cell faces (represented by non capital letters l and r). This means that very often numerical interpolation schemes are necessary to obtain cell face values from cell center quantities. One way of achieving this at a particular surface, say face r, is by evaluating the linear interpolation between R and P cell center values, using a method known as central differencing scheme (CDS). The CDS scheme is second order accurate but does not take into account the direction of the flow, and therefore is not able to recognise the strength of convection relative to diffusion. This can cause instability for the solution in a highly convective flow from L to R, because the value at cell face r will be more dependent on the upstream quantity than on its downstream counterpart. A remedy for that would be the use of the upstream differencing scheme, in which the cell face value is assumed to be equal to the upstream value ( $\Phi_l = \Phi_L$  and  $\Phi_r = \Phi_P$ ). Although the upwind scheme is more stable than the CDS scheme for considering the flow direction, specially in high speed flows, it introduces considerable amounts of false diffusion to the solution due to its first order accuracy character. Thus, in order to find a compromise between the accuracy of CDS schemes and the stability of upwind schemes, total volume diminishing (TVD) schemes were developed. Also known as high-order schemes, the idea of their implementation is to define several interpolation points between P and R, so that the value at cell face r can be obtained by a weighting interpolation process, assigning higher importance to the upstream contributions. Several types of TVD schemes have been developed and successfully implemented in CFD codes. For a more in depth discussion on CDS, upwind, and TVD schemes, please refer to Versteeg and Malalasekera (2007), Patankar and Spalding (1972), Ferziger and Peric (2002).

Moving on with the discretization process, a common practical approach in CFD codes is the representation of the source/sink term  $S_\Phi$  per unit area in terms of the dependent variable  $\Phi$ , as shown

in equation (72):

$$\bar{S}_\Phi \Delta x = S_u + S_p \Phi_P \quad (72)$$

where  $S_u$  and  $S_p$  are the linearised terms of the source/sink.

Therefore, the expression for the steady-state convective-diffusive transport of quantity  $\Phi$  throughout the domain can be written as in equation (73):

$$F_r \Phi_r - F_l \Phi_l = D_r(\Phi_R - \Phi_P) - D_l(\Phi_P - \Phi_L) + S_u + S_p \Phi_P \quad (73)$$

Considering CDS differencing scheme, the above equation can be rearranged as in equation (74):

$$\Phi_P \left[ \frac{F_r}{2} - \frac{F_l}{2} + (D_r + D_l) - S_p \right] = \Phi_L \left[ \frac{F_l}{2} + D_l \right] + \Phi_R \left[ -\frac{F_r}{2} + D_r \right] + S_u \quad (74)$$

for which the  $a_L$ ,  $a_R$ , and  $a_P$  coefficients can be defined for the left, right, and cell contributions, as shown in Table 8, resulting in equation (75):

Table 8 – Discretized equation coefficients

$a_P$	$a_L$	$a_R$
$\frac{F_r}{2} - \frac{F_l}{2} + (D_r + D_l) - S_p$	$\frac{F_l}{2} + D_l$	$-\frac{F_r}{2} + D_r$

Source: Elaborated by the author.

$$a_p \Phi_P = a_L \Phi_L + a_R \Phi_R + S_u \quad (75)$$

The one-dimensional formulation described above can be easily extended to three-dimensions adding further coefficients to the fluxes in other directions, using appropriate subscripts, such as U(up) and D(down), and F(front) and B(back), as shown in equation (76):

$$a_p \Phi_P = a_L \Phi_L + a_R \Phi_R + a_D \Phi_D + a_U \Phi_U + a_B \Phi_B + a_F \Phi_F + S_u \quad (76)$$

Therefore, a computational domain composed of N grid elements will require the solution of a system of N algebraic equations of the form shown in equation (77), in which the subscript i represents the grid elements and nb refers to each of the M neighbor elements of i. For this purpose, different kinds of linear equation matrix solvers can be employed, using distinct numerical techniques, such as preconditioning and matrix decomposition, to achieve better efficiency.

$$a_i \Phi_i = \sum_{nb=1}^M a_{nb} \Phi_{nb} + S_{u_i} \quad (77)$$

Finally, since engine flows are strongly transient phenomena, it is necessary to account for changes of  $\Phi$  in time, as described by the accumulation term shown in discretized form in equation (78):

$$\frac{\partial}{\partial t}(\rho \Phi) = \rho \Phi \frac{\Delta V}{\Delta t} \quad (78)$$

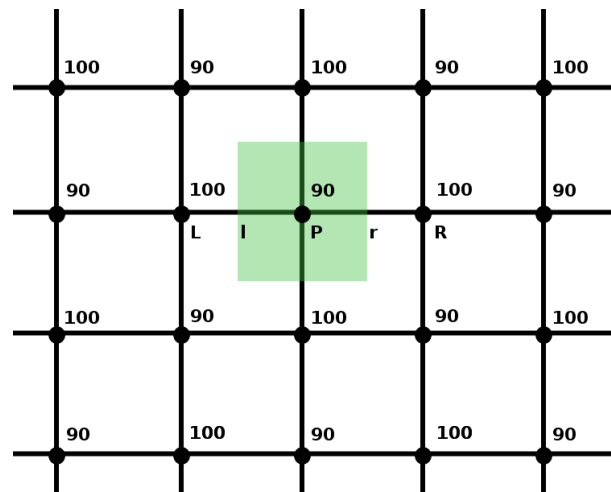
In order to do that, a numerical time integration scheme must be employed, which can be done in two different forms. On the one hand, the quantity  $\Phi$  can be evaluated at the current time-step explicitly, i.e. only information from the previous time-step is used for its evaluation. Or, on the other hand,  $\Phi$  can be evaluated implicitly, which means that information from both current and previous time-steps are used, thus requiring the use of an iterative procedure. Both strategies have advantages and disadvantages, with explicit schemes being normally more efficient than implicit ones, whereas implicit schemes are commonly more stable than their explicit counterpart. As done for the interpolation techniques in CFD, the more in depth description of the time integration strategies are left in charge of the reader, being here suggested some references in the literature, such as Boyce and DiPrima (2005), Ferziger and Peric (2002), and Versteeg and Malalasekera (2007).

After the description of the basic discretization process of the equation for the convective-diffusive transport of the scalar  $\Phi$ , it is time to describe how it can be employed on the Navier-Stokes equations presented in the last section. The discretization procedure itself can be generalized for all quantities of interest simply by swapping  $\Phi$  for the desired variable, but the solution method may vary from equation to equation, due to differences in complexity in the underlying physics. For example, regarding the mass conservation equation, the solution procedure is quite simple, requiring only the evaluation of convective fluxes of density ( $\rho$ ) through all control volume element faces and imposing a null value for their overall balance (unless there is a source or sink of mass inside the cell element, for which case an expression will have to be assigned to the source/sink term). For a transient solution, the accumulation of  $\rho$  in the domain must also be taken into account, which can be done by means of either an explicit or implicit time integration method and by adding the discretized accumulation term  $\frac{\rho\Delta V}{\Delta t}$ . This process will produce an equation easy to be linearised, which can be solved by the linear matrix solver at hand. Moving on to the energy equation, since the total enthalpy  $h_t$  is a scalar, the procedure will be similar to the one shown above for the transport of the generic scalar  $\Phi$ , although additional terms may be present to consider different types of work being expended on the fluid, as for example by the action of viscous dissipation. Finally, the greatest challenge is the solution regarding the momentum equation due to its strong non-linear nature, with each velocity component  $u$ ,  $v$ , and  $w$  being coupled to each other in each of the velocity component equations. Furthermore, the pressure field itself is part of the numerical solution, which in the case of compressible flows can be obtained by the evaluation of a equation of state ( $p = f(\rho, T)$ ), with the value of density being transported by the continuity equation and temperature being obtained by the solution of the energy equation. However, as shown in Versteeg and Malalasekera (2007), the pressure-velocity coupling is much stronger than the non-linear velocity coupling in the momentum equation and a stable converging solution may be hard to achieve even if the continuity equation is satisfied. In order to overcome these issues associated with the non-linear velocities and pressure linkage of the momentum equation, Patankar (1980) developed the Semi-Implicit Method for Pressure Linked Equations (SIMPLE) algorithm, which aims at obtaining a solution for the velocity field.

The SIMPLE algorithm is an iterative procedure that works by guessing initial velocity and pressure fields, then correcting both properties at each iteration until a final reasonably converged solution is achieved. This is done by correcting the pressure field by an expression deduced from the continuity

equation during every iterative step. Following that, within the same step the velocity field must be re-evaluated by coupling the solution for the convective-diffusive transport of all velocity components to the corrected pressure gradient term in each direction. Again, the convective-diffusive transport of each velocity component can be done using the discretization procedure described above by replacing  $\Phi$  with  $u$ ,  $v$ , or  $w$ . However, unless special meshing techniques are employed, velocity and pressure data cannot be stored in the same grid location, because this can result in inconsistent evaluations of the pressure gradient field. In order to demonstrate that, it is assumed that a “checker-board” pressure field was achieved in a certain time step, as shown in Figure 80. This means that, for the control volume shown in the illustration, if one calculates the pressure influence in both velocity component equations  $u$  and  $v$ , a null pressure gradient will be obtained, as shown in equations (79) and (80). This is of course a non-physical behaviour and should be avoided.

Figure 80 – Conventional Grid



Source: Elaborated by the author.

$$\frac{p_r - p_l}{\delta x} = \frac{p_R - p_L}{2\delta x} = 0 \quad (79)$$

$$\frac{p_u - p_d}{\delta y} = \frac{p_U - p_D}{2\delta y} = 0 \quad (80)$$

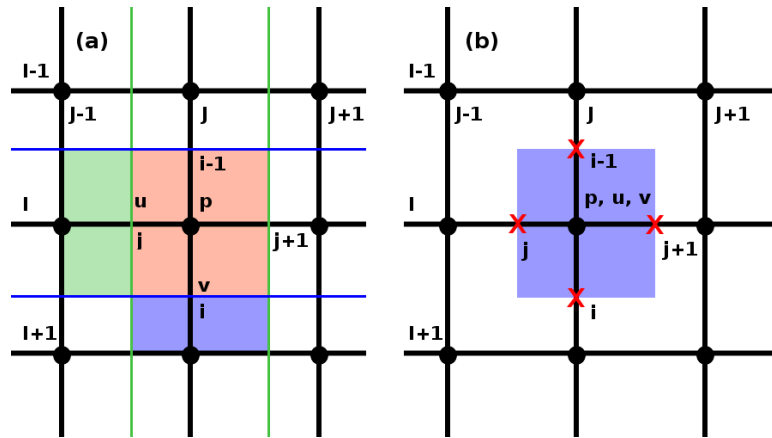
One way to circumvent the non-physical behaviour described above is the use of a staggered grid, as shown in Figure 81-a for a two-dimensional domain. That means that at one particular cell (red region in the figure) the pressure  $p$  is stored at the node (black circle), the horizontal velocity component  $u$  is stored at the face on the left side of the node, and the vertical velocity component  $v$  is located at the face under the node. While this approach successfully avoids the aforementioned issue, its implementation is very inconvenient in complex domains, such as the object of study of this work (i.e. the combustion chamber of an internal combustion engine). Therefore, in this work the so called collocated mesh strategy is preferred to deal with the problem associated with the velocity-pressure coupling. In a collocated grid, velocity and pressure data are stored at the cell node, as shown in Figure 81-b, using an appropriate choice of interpolation schemes to the cell faces to minimize problems related to the checker-board field (VERSTEEG; MALALASEKERA, 2007).

Independent of the meshing approach used, the SIMPLE solution procedure starts with the calculation of the u and v velocity components from the guessed pressure field, as shown in equation (81) for u and equation (82) for v, where the star sign means guessed values.

$$a_{I,j}u_{I,j}^* = \sum_{nb=1}^M a_{nb}u_{nb}^* - (p_{I,J}^* - p_{I,J-1}^*)A_{I,j} + \bar{S}u\Delta V_{I,j} \quad (81)$$

$$a_{i,J}v_{i,J}^* = \sum_{nb=1}^M a_{nb}v_{nb}^* - (p_{I,J}^* - p_{I+1,J}^*)A_{i,J} + \bar{S}v\Delta V_{i,J} \quad (82)$$

Figure 81 – Staggered Grid vs Collocated Grid



Source: Elaborated by the author.

This will result in a guessed velocity field, which very likely will not represent the correct solution for velocity, since the final pressure solution was not yet achieved. In order to obtain the final solution for both pressure and velocity, corrections must be added to all fields in an iterative procedure until convergence is accomplished. The correction expressions for pressure and both velocity components are shown in equations (83), (84), and (85):

$$p = p^* + p' \quad (83)$$

$$u = u^* + u' \quad (84)$$

$$v = v^* + v' \quad (85)$$

for which correction terms  $p'$ ,  $u'$ , and  $v'$  are added to the guessed solutions of each field.

The correction terms for u and v equations are obtained from the subtraction of two subsequent u and v fields, as shown in equations (86) and (87):

$$a_{I,j}u_{I,j}' = \sum_{nb=1}^M a_{nb}u_{nb}' - (p_{I,J}' - p_{I,J-1}')A_{I,j} \quad (86)$$

$$a_{i,J}v_{i,J'} = \sum_{nb=1}^M a_{nb}v_{nb}' - (p_{I,J'} - p_{I+1,J'})A_{i,J} \quad (87)$$

The SIMPLE algorithm then assumes that the influence of the neighbor elements on each cell is negligible compared to the influence of pressure, thus the corrections for  $u$  and  $v$  are shown in equations (88) and (89):

$$u_{I,j}' = d_{I,j}(p_{I,J-1}' - p_{I,J}')A_{I,j} \quad (88)$$

$$v_{i,J'} = d_{i,J}(p_{I+1,J'} - p_{I,J}')A_{i,J} \quad (89)$$

where the terms  $d$  refer to the division of the cell face area by the convection-diffusion coefficients, such as  $d_{I,j} = \frac{A_{I,j}}{a_{I,j}}$  and  $d_{i,J} = \frac{A_{i,J}}{a_{i,J}}$ .

For the derivation of the pressure correction, the discretized continuity equation (90) is used, employing the corrected velocity fields obtained above.

$$[(\rho u A)_{I,j+1} - (\rho u A)_{I,j}] + [(\rho v A)_{i-1,J} - (\rho v A)_{i,J}] = 0 \quad (90)$$

The resulting expression is shown in equation (91), where  $a_{I,J} = a_{I,j} + a_{I,j+1} + a_{i,J} + a_{i-1,J}$  and all coefficients are summarized in Table 9:

$$a_{I,J}p_{I,J}' = a_{I,j}p_{I,j}' + a_{I,j+1}p_{I,j+1}' + a_{i,J}p_{i,J}' + a_{i-1,J}p_{i-1,J}' + b_{I,J}' \quad (91)$$

Table 9 – Pressure correction equation coefficients

$a_{I,j}$	$a_{I,j+1}$	$a_{i,J}$	$a_{i-1,J}$	$b_{I,J}'$
$(\rho dA)_{I,j}$	$(\rho dA)_{I,j+1}$	$(\rho dA)_{i,J}$	$(\rho dA)_{i-1,J}$	$(\rho u^* A)_{I,j+1} - (\rho u^* A)_{I,j} + (\rho v^* A)_{i-1,J} - (\rho v^* A)_{i,J}$

Source: Elaborated by the author.

The term “b” in the pressure correction equation represents a mass source in the continuity equation, and can be thus considered a convergence criterion, i.e. the iteration procedure should stop when this term reaches a value below a certain tolerance. Finally, the full SIMPLE algorithm can be listed as follows:

1. Guess a pressure field  $p^*$ ;
2. Solve the discretized equations for  $u^*$ ,  $v^*$ , and  $w^*$  from the guessed pressure field  $p^*$ ;
3. Solve the pressure correction equation ( $p'$ ) and correct the pressure field with equation (83);
4. Correct velocity fields with the corrected pressure equation;
5. Solve transport equations for all other variables;

6. Check if convergence was achieved. If not, treat the current pressure field as the guessed field and start over from 2;

The procedure presented above can be used to obtain a steady-state solution for pressure, velocity and other properties, but it can be extended to transient solutions if a numerical time integration approach is adopted. Other similar numerical methods have been proposed in the literature with similar goals to the SIMPLE procedure, among which the PISO (Pressure Implicit with Splitting of Operator) algorithm developed by Issa (1986) can be highlighted. The idea behind PISO is to split the pressure corrector in two or more steps, so that the first one will create a conservative field and the second or following will stabilize the pressure field. This is again possible due to the linear pressure-velocity correction coupling being much stronger than the non-linear coupling present in the discretised momentum equations for  $u^*$ ,  $v^*$  and  $w^*$ , allowing a bigger number of pressure corrections than the SIMPLE procedure at small time steps. Finally, in this work a hybrid between PISO and SIMPLE algorithms is employed (OPENFOAM, 2017). The so-called PIMPLE algorithm is basically the PISO algorithm with the addition of eventual outer correction loops of the variables over each time step using the last iteration final value as initial guess for the next iteration, such as in the SIMPLE method. Furthermore, the exact closure for the pressure equation is obtained by the pressure-velocity-density coupling for flows at arbitrary Mach numbers proposed by Ferziger and Peric (2002).

## APPENDIX B – NUMERICAL SETUP OF SIMULATIONS

### B.1 COLD-FLOW ENGINE CASES

The most difficult part in setting up the engine simulations of this work is the necessity of working in several case folders because of the use of the mapping strategy to deal with cell deformation. This deformation is caused by the movement of the piston and valves during the engine's operational cycle; if not dealt with, this will cause the cell quality to decrease drastically and the simulation to crash. The remedy adopted for that is a remeshing procedure every 5 CAD, mapping the results from the older to the newer mesh. In order to speed up the process, all the meshes used throughout the cycle are previously created and stored in a directory called "Stock". Then a shell script is employed to create all the case folders from the beginning of the intake stroke (here defined as -360 degrees before TDC) until the end of the exhaust stroke (355 CAD for a complete 720 degrees revolution of a four-stroke cycle), as illustrated in Figure 82. At the same time each of these folders is generated, all the necessary sub-directories are created, as well as boundary conditions, sub-model dictionaries, and mesh files are copied to each of them. For its specific purpose, this shell script is called "caseSetup".

Figure 82 – Engine simulation folders



Source: Elaborated by the author.

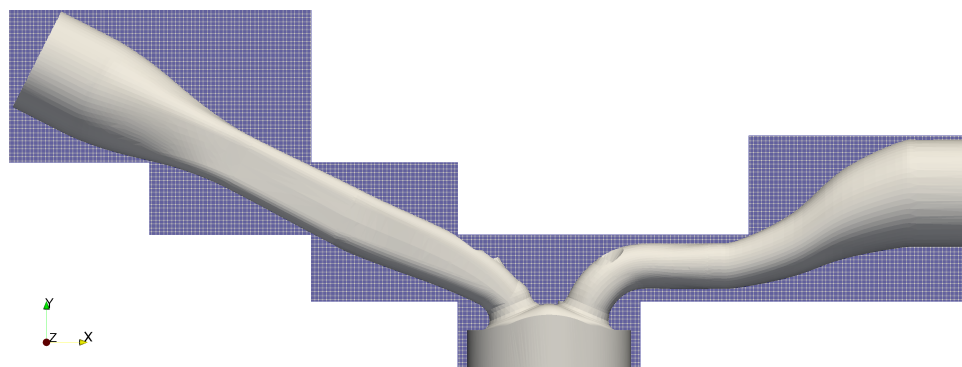
However, prior to the execution of the "caseSetup" script, all grids for each 5 CAD interval must have been already generated. In order to do so, another script is used to automatically call the OpenFOAM native mesh generation utilities *blockMesh* and *snappyHexMesh*. The former creates an initial domain composed of only perfectly orthogonal hexahedral cells, which must encompass the desired computational domain represented by an STL<sup>1</sup> geometry file, defining the initial cell resolution. The latter creates the final grid by a process of iteratively refining and splitting the initial mesh created with *blockMesh* and morphing the resulting split-hex mesh to the surface of the provided geometry. This work-flow is illustrated in Figure 83, for the initial domain creation, and Figure 84, showing the final mesh generated with *snappyHexMesh*. The final mesh has a resolution of 0.7 mm for cells away from the walls in the cylinder, but due to the special refinement, cell sizes can be as small as 0.35 mm around the spark-plug region and 0.175 mm at the valve seat (during valve closing), similarly as in Janas (2017), Ribeiro (2018). The cell count is 4.4 million at BDC and 2.75 million at TDC.

In a similar way to the "caseSetup" script, the "makeMesh" script loops from -360 CAD to 360 CAD, copying all necessary files (such as the STL geometry file), calling mesh generation utilities, and saving their results in the "Stock" folder, for each 5 CAD interval. The *snappyHexMesh* utility can run in parallel and it allows special refinement in regions of high geometric complexity, such as the

<sup>1</sup> STL is an abbreviation for "stereolithography", which is a geometry file format that describes a surface with adjacent triangles. The file contains a list of all triangles (unit normal and vertices) necessary to represent the geometry

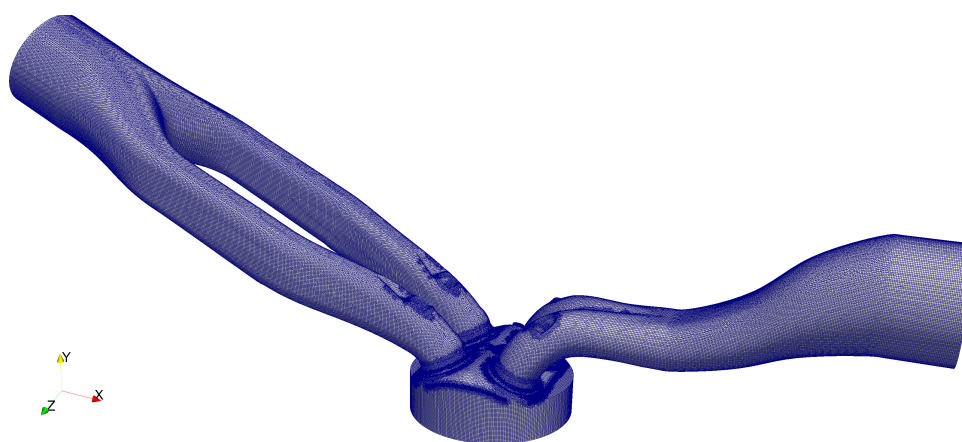


Figure 83 – Mesh generation process at 45 bTDC - initial domain generation



Source: Elaborated by the author.

Figure 84 – Mesh generation process at 45 bTDC - final mesh



Source: Elaborated by the author.

piston head, and/or regions where high pressure/velocity gradients are expected. Moreover, the closure of the valves is done by the generation of internal walls between valves and cylinder head.

After all grids have been generated, a third script, "runCase", can be executed to run the engine simulation in parallel, by decomposing the mesh in different processors, calling the engine solver, and mapping the results of one mesh to the next every 5 CAD interval. Throughout the simulation duration, the piston position "p" changes according to the reciprocating piston equation, shown in equation (92), and the valves follow a profile provided by experimental measurements. The timing for opening and closure of valves is presented in Table 10, and a summary of boundary conditions used in the simulation is provided in Table 11. In regard to the boundary conditions, inlet and outlet pressure values are inputted with experimental profiles by piezoelectric sensors, whereas for velocity a zeroGradient condition is used for the the same boundaries. Moreover, wall temperatures are also based on measured data. Finally, total variation diminishing (TVD) schemes are used for the convective terms of the governing equations to avoid numerical instability, as well as a second order implicit backward scheme is used for the time integration with an imposed maximum Courant-Friedrich-Lewy (CFL) number of 0.5. For the numerical solution, PBiCG matrix solvers are used for almost all flow properties, except for the case of density, for which a diagonal solver is used, and the cell motion, for

which a GAMG solver is employed.

$$p = p_{TDC} - \left( (L_C + S/2) - S * \cos(\alpha)/2 - \sqrt{L_C^2 - (S * \sin(\alpha)/2)^2} \right) \quad (92)$$

In the previous equation  $p_{TDC}$  is the piston position at TDC,  $L_C$  is the connecting rod length,  $S$  is the piston stroke, and  $\alpha$  represents the crank angle degree.

Table 10 – Valve timing

Intake valves open	35 CAD before TDC
Intake valves close	125 CAD before TDC
Exhaust valves open	105 CAD after TDC
Exhaust valves close	15 CAD after TDC

Source: Elaborated by the author.

Table 11 – Summary of boundary conditions

Boundary	U	p	T
Inlet	zeroGradient	exp. profile	295 K
Outlet	zeroGradient	exp. profile	zeroGradient
Intake walls	fixedValue = 0	zeroGradient	295 K
Exhaust walls	fixedValue = 0	zeroGradient	333 K
Cylinder walls	fixedValue = 0	zeroGradient	333 K
Piston	movingWallVelocity	zeroGradient	333 K
Intake Valves	movingWallVelocity	zeroGradient	300 K
Exhaust Valves	movingWallVelocity	zeroGradient	333 K

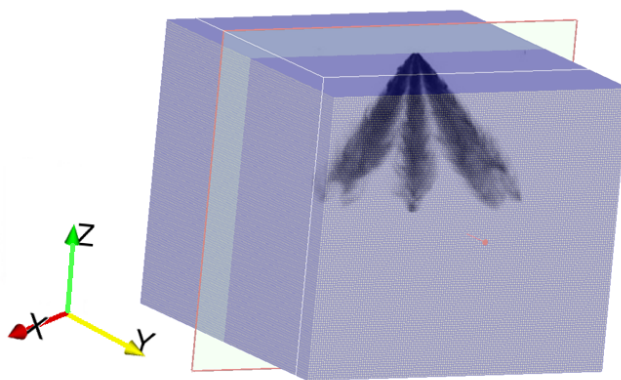
Source: Elaborated by the author.

## B.2 FUEL-SPRAY CASES

The setup for simulating both ECN Spray G and Spray A validation cases is quite simpler than the one required for the engine simulations shown in the previous section, since the injection is done in a constant volume combustion chamber, and thus, no moving mesh is necessary. Therefore, the meshing for the spray simulations presented in this section is performed with the OpenFOAM *blockMesh* utility, generating a computational domain composed of perfect hexahedrons, as shown in Figure 85 for the Spray-G case. The domain dimensions are 140 mm in both radial (x) and transverse (y) directions, and 120 mm in the axial (z) direction, totalling 2.35 million elements for a cell resolution of 0.5 mm. For the Spray A case, since the injector has only one axially oriented hole, the domain dimensions were changed to 30 mm in both radial and transverse directions, and 140 mm in the axial direction, making the cell count only slightly higher than 1 million (1,008,000 cells). Regarding the numerical schemes, central differential is used for the convective term in the momentum equation, whereas TVD schemes are used for the further convective terms. Moreover, a second order backward implicit time integration scheme is used for marching in time with a CFL number of 0.1.

The boundary conditions for pressure, temperature, density and species fields are kept as zero gradient at the walls, whereas for velocity non-slip conditions are used. In regard to the initial

Figure 85 – Spray G simulation computational mesh



Source: Elaborated by the author.

conditions, the domain is initialized according to the definitions of the ECN community, which specifies the exact mass fraction of each of the species in the non-reactive gaseous environment, as well as the initial gas temperature of 573 K (for Spray G) or 900 K (for Spray A), the gas pressure of 6 bar (Spray G) or 60 bar (Spray A), and the gas density of  $3.5 \text{ kg/m}^3$  (Spray G) and  $22.8 \text{ kg/m}^3$  (Spray-A), as summarized in the previous chapter (validation data). Moving on to the boundary conditions of the liquid phase, points are defined along an axial plane at the top of the domain (8 for Spray G and 1 for Spray A), in order to mimic the position of the injector nozzle exits. From these points, liquid jets are simulated by the injection of Lagrangian parcels containing iso-octane (Spray G) or n-heptane (Spray A). For the Spray G injector the fuel jets are injected at an angle of 37 degrees relative to the axial axis z, defined here as the plume direction angle, and for the Spray A injector the jet is injected along the axial direction. Furthermore, the temperature at which parcels are injected is kept constant at 363 K and droplets follow a profile deduced from the mass flow rate obtained experimentally by ECN contributors (ECN, 2018). Initial droplet diameters follow a Rosin-Rammler distribution with maximum value similar to the nozzle hole diameter of  $165 \mu\text{m}$  and a minimum value of  $1 \mu\text{m}$  for the Spray G case and  $90 \mu\text{m}/1 \mu\text{m}$  for the Spray A case, being the droplets subject to breakup after injection as described by the Reitz (1987) model. The total iso-octane (Spray G) mass injected sums up to 10 mg (1.25 mg per hole) within an injection duration of  $780 \mu\text{s}$ , while the iso-octane/n-heptane (Spray A) mass sums up to 6.92 mg within an injection duration of 3 ms. Heat transfer between droplet and gas is modelled with the Ranz and Marshall (1952) model, whereas evaporation is modelled with the Zuo, Gomes and Rutland (2000) model. Furthermore, the reaction mechanism of Zhou et al. (2016) is employed to simulate the reactions taking place in the Spray A case, representing the combustion among 43 species in 144 reactions. The initial concentration of species is obtained by means of equilibrium rate calculations performed with the Cantera code (GOODWIN; MOFFAT; SPETH, 2017). Finally, experimental data in regard to ignition delay and flame lift-off length are available for the injection of iso-octane/n-heptane blends (also called primary reference fuels - PRF), with octane number ranging from 0 to 100 in steps of 20. In this work, a total of three preliminary simulations have been conducted so far, for octane numbers 0, 60, and 100.

### B.3 FIRED ENGINE CASES

The setup for the fired engine cases contains elements of the different setups described above, since once again a moving grid will be necessary to consider the movement of valves and piston, with the addition of the Lagrangian-Eulerian approach together with a reaction kinetics model to simulate fuel injection and flame propagation. For the sake of brevity, only the parameters specific to the Darmstadt engine under fired operation will be shown here, whereas information already shown in previous sections on how to perform the meshing/solution procedure and information concerning the application of some of the boundary conditions will be here suppressed.

Regarding the direct injection boundary conditions, information is provided in Table 12. As shown, engine speed is kept constant at 800 rpm when at 75 CAD bTDC iso-octane is injected into the cylinder in the form of a hollow-cone spray with total angle of 105°. The injected mass for the motored case was 2.9 mg, whereas for the fired case 13 mg of iso-octane were injected. Furthermore, the injection pressure of this particular injector is 18 MPa, and the fuel is injected at a temperature of 363 K.

Table 12 – Spray-guided Darmstadt engine hollow-cone injection parameters

Injection type	hollow-cone spray
Engine speed	800 rpm
Fuel	iso-octane
Spray angle	105°
Start of injection	75 CAD bTDC
Injected mass	2.9 mg (motored), 13 mg (fired)
Injection pressure / temperature	18 MPa / 363 K

Source: Elaborated by the author.

Concerning the reaction mechanism and combustion model used, more information is given in Table 13. As a first approach, a single equation reaction mechanism was used, in which the oxidation of iso-octane results in the formation of carbon dioxide and water vapour. The reaction term closure in the energy equation and species transport equation is accomplished with the Partially Stirred Reactor (PaSR) model, setting the spark timing at 19 CAD bTDC with a duration of 5 CAD. The spark is simulated simply by imposing the adiabatic flame temperature of iso-octane in a spherical region with diameter 2 mm at the spark position.

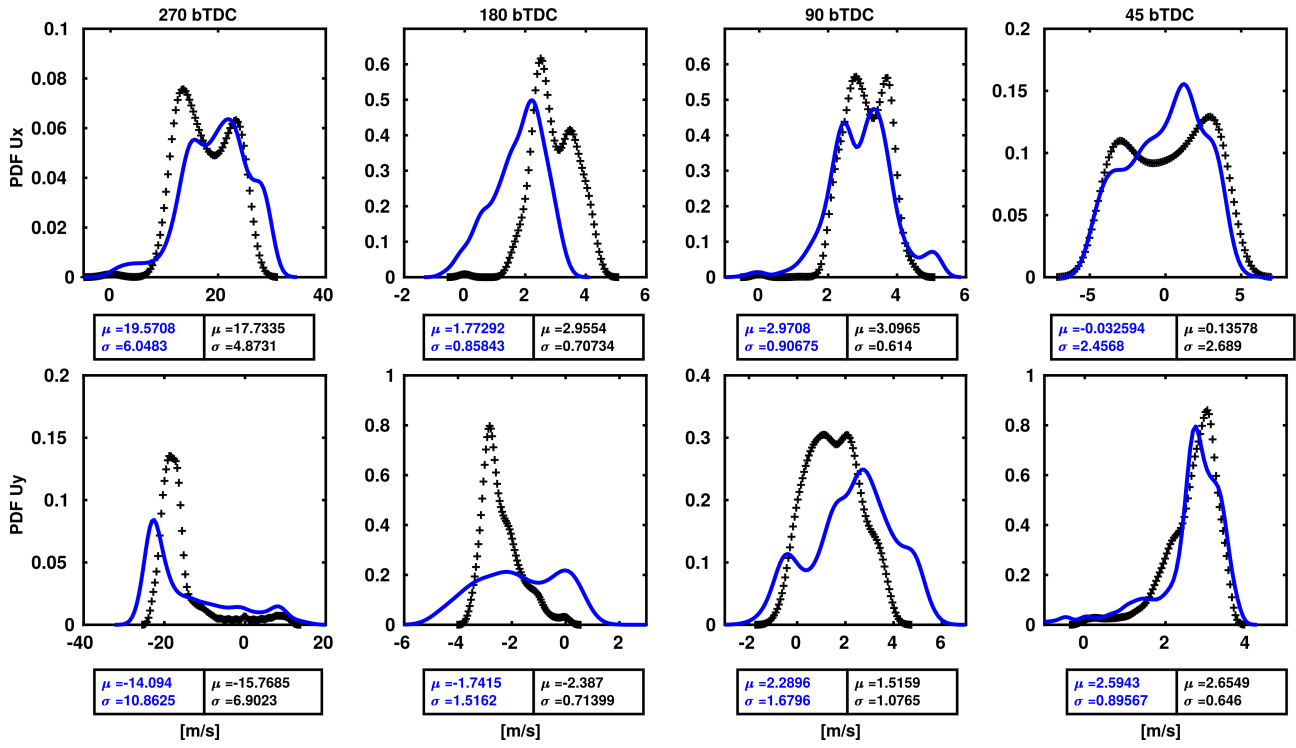
Table 13 – Information on the GDI engine fired case

Reaction mechanism	single equation ( $C_8H_{18} + 12.5O_2 \rightarrow 8CO_2 + 9H_2O$ )
Combustion model	Partially Stirred Reactor (PaSR) model
Spark timing	19 CAD bTDC
Spark duration	5 CAD
Kinetics solver	CHEMKIN

Source: Elaborated by the author.

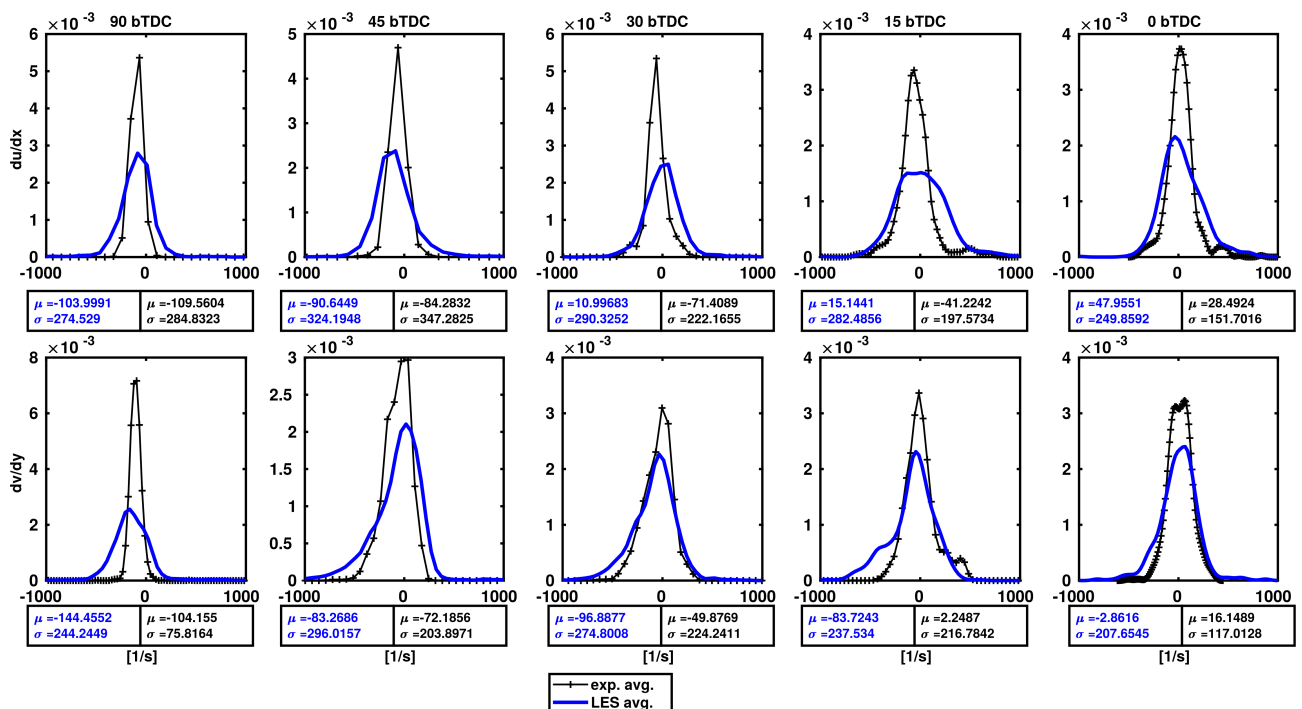
APPENDIX C – ADDITIONAL RESULTS

Figure 86 – Velocity PDFs at 270, 180, 90, and 45 CAD bTDC



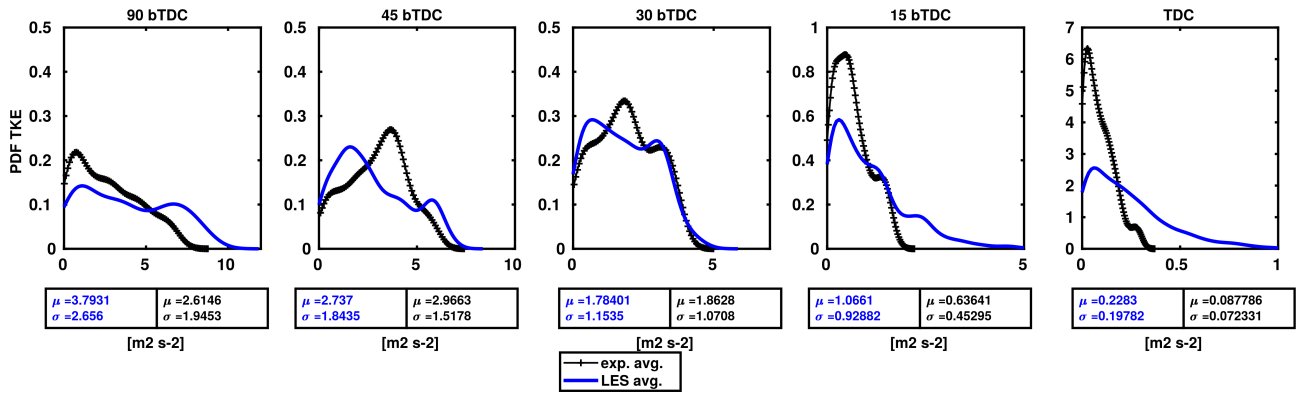
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 87 – PDF of velocity gradients



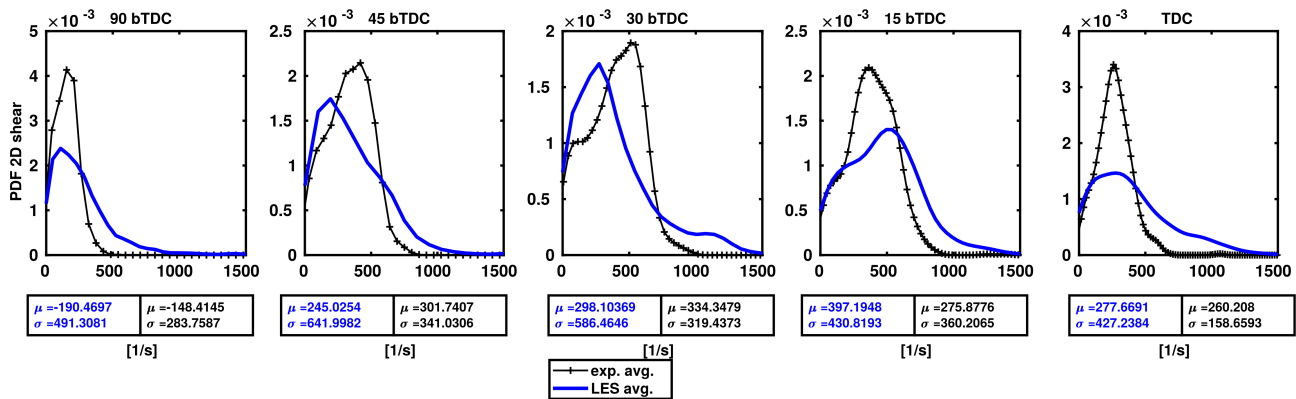
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 88 – PDF of turbulent kinetic energy



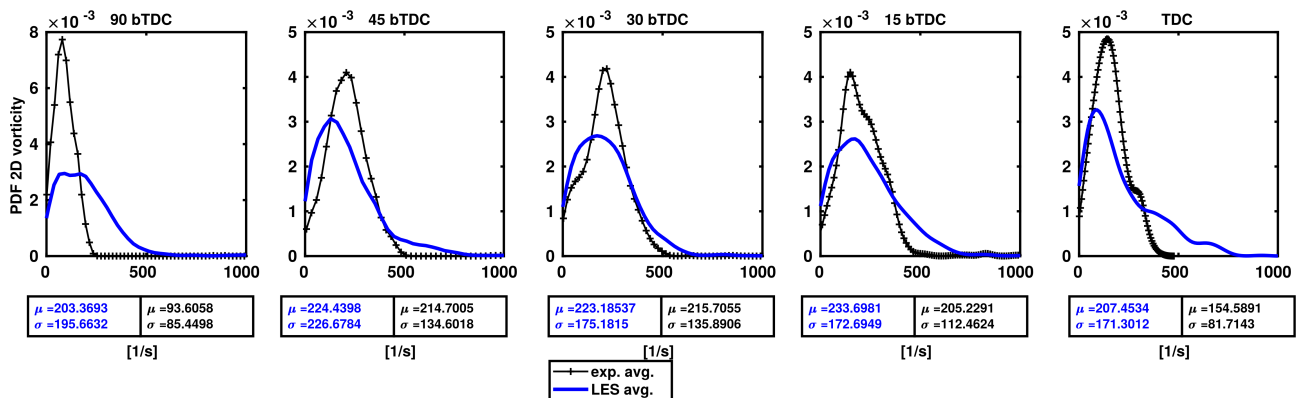
Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 89 – PDF of 2D shear



Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

Figure 90 – PDF of 2D vorticity



Source: LES (Elaborated by the author). Experiment Freudenhammer et al. (2015).

## Frame 2 – OFICENet Training log

Layer (type)	Output Shape	Param #
dense_11 (Dense)	(None, 20)	140
dense_12 (Dense)	(None, 20)	420
dense_13 (Dense)	(None, 20)	420
dense_14 (Dense)	(None, 20)	420
dense_15 (Dense)	(None, 1)	21

Total params: 1,421, Trainable params: 1,421, Non-trainable params: 0  
 Train on 3545989 samples, validate on 886498 samples  
 Epoch 1/1000  
 3545989/3545989 [=====] – 79s 22us/step  
 loss: 0.0171 – mean\_absolute\_error: 0.0171  
 val\_loss: 0.0172 – val\_mean\_absolute\_error: 0.0172  
 Epoch 00001: val\_loss improved **from** inf to 0.01723,  
 saving model to Weights-001--0.01723.hdf5  
 Epoch 2/1000  
 3545989/3545989 [=====] – 77s 22us/step  
 loss: 0.0171 – mean\_absolute\_error: 0.0171  
 val\_loss: 0.0171 – val\_mean\_absolute\_error: 0.0171  
 Epoch 00002: val\_loss improved **from** 0.01723 to 0.01712,  
 saving model to Weights-002--0.01712.hdf5  
 Epoch 3/1000  
 3545989/3545989 [=====] – 77s 22us/step  
 loss: 0.0171 – mean\_absolute\_error: 0.0171  
 val\_loss: 0.0170 – val\_mean\_absolute\_error: 0.0170  
 Epoch 00003: val\_loss improved **from** 0.01712 to 0.01701,  
 saving model to Weights-003--0.01701.hdf5  
 Epoch 4/1000  
 3545989/3545989 [=====] – 77s 22us/step  
 loss: 0.0171 – mean\_absolute\_error: 0.0171  
 val\_loss: 0.0171 – val\_mean\_absolute\_error: 0.0171  
 Epoch 00004: val\_loss did **not** improve **from** 0.01701  
 Epoch 5/1000  
 3545989/3545989 [=====] – 78s 22us/step  
 loss: 0.0171 – mean\_absolute\_error: 0.0171  
 val\_loss: 0.0170 – val\_mean\_absolute\_error: 0.0170

```
47 Epoch 00005: val_loss did not improve from 0.01701
48 Epoch 6/1000
49 3545989/3545989 [=====] - 77s 22us/step
50 loss: 0.0171 - mean_absolute_error: 0.0171
51 val_loss: 0.0171 - val_mean_absolute_error: 0.0171
52
53 Epoch 00006: val_loss did not improve from 0.01701
54 Epoch 7/1000
55 3545989/3545989 [=====] - 77s 22us/step
56 loss: 0.0171 - mean_absolute_error: 0.0171
57 val_loss: 0.0170 - val_mean_absolute_error: 0.0170
58
59 Epoch 00007: val_loss improved from 0.01701 to 0.01699,
60 saving model to Weights-007--0.01699.hdf5
61 Epoch 8/1000
62 3545989/3545989 [=====] - 78s 22us/step
63 loss: 0.0171 - mean_absolute_error: 0.0171
64 val_loss: 0.0178 - val_mean_absolute_error: 0.0178
65
66 Epoch 00008: val_loss did not improve from 0.01699
67 Epoch 9/1000
68 3545989/3545989 [=====] - 77s 22us/step
69 loss: 0.0171 - mean_absolute_error: 0.0171
70 val_loss: 0.0177 - val_mean_absolute_error: 0.0177
71
72 Epoch 00009: val_loss did not improve from 0.01699
73 Epoch 10/1000
74 3545989/3545989 [=====] - 77s 22us/step
75 loss: 0.0170 - mean_absolute_error: 0.0170
76 val_loss: 0.0169 - val_mean_absolute_error: 0.0169
77
78 Epoch 991/1000
79 3545989/3545989 [=====] - 79s 22us/step
80 loss: 0.001425 - mean_absolute_error: 0.00161
81 val_loss: 0.00162 - val_mean_absolute_error: 0.00162
82
83 Epoch 00992: val_loss did not improve from 0.00162
84 Epoch 992/1000
85 3545989/3545989 [=====] - 77s 22us/step
86 loss: 0.001424 - mean_absolute_error: 0.00161
87 val_loss: 0.00161 - val_mean_absolute_error: 0.00161
88
89 Epoch 00993: val_loss improved from 0.01723 to 0.01712,
90 saving model to Weights-002--0.01712.hdf5
91 Epoch 993/1000
92 3545989/3545989 [=====] - 77s 22us/step
93 loss: 0.001424 - mean_absolute_error: 0.00161
94 val_loss: 0.00160 - val_mean_absolute_error: 0.00160
95
```



```

96 Epoch 00994: val_loss improved from 0.001612 to 0.001601,
97 saving model to Weights-003--0.01701.hdf5
98 Epoch 994/1000
99 3545989/3545989 [=====] - 77s 22us/step
100 loss: 0.001424 - mean_absolute_error: 0.00161
101 val_loss: 0.00161 - val_mean_absolute_error: 0.00161
102
103 Epoch 00995: val_loss did not improve from 0.001701
104 Epoch 995/1000
105 3545989/3545989 [=====] - 78s 22us/step
106 loss: 0.001424 - mean_absolute_error: 0.00161
107 val_loss: 0.00160 - val_mean_absolute_error: 0.00160
108
109 Epoch 00996: val_loss did not improve from 0.001601
110 Epoch 996/1000
111 3545989/3545989 [=====] - 77s 22us/step
112 loss: 0.001424 - mean_absolute_error: 0.00161
113 val_loss: 0.00161 - val_mean_absolute_error: 0.00161
114
115 Epoch 00997: val_loss did not improve from 0.001701
116 Epoch 997/1000
117 3545989/3545989 [=====] - 77s 22us/step
118 loss: 0.001424 - mean_absolute_error: 0.00161
119 val_loss: 0.00160 - val_mean_absolute_error: 0.00160
120
121 Epoch 00998: val_loss improved from 0.001601 to 0.01699,
122 saving model to Weights-007--0.01699.hdf5
123 Epoch 998/1000
124 3545989/3545989 [=====] - 78s 22us/step
125 loss: 0.001424 - mean_absolute_error: 0.00161
126 val_loss: 0.00168 - val_mean_absolute_error: 0.00168
127
128 Epoch 00999: val_loss did not improve from 0.001699
129 Epoch 999/1000
130 3545989/3545989 [=====] - 77s 22us/step
131 loss: 0.001424 - mean_absolute_error: 0.00161
132 val_loss: 0.00167 - val_mean_absolute_error: 0.00167
133
134 Epoch 01000: val_loss did not improve from 0.001699
135 Epoch 1000/1000
136 3545989/3545989 [=====] - 77s 22us/step
137 loss: 0.001424 - mean_absolute_error: 0.00170
138 val_loss: 0.00169 - val_mean_absolute_error: 0.00169
139
140 Epoch 01000: val_loss improved from 0.001699 to 0.001695,
141 saving model to Weights-010--0.001695.hdf5

```

---

Source: Elaborated by the author.

## ANNEX A – SIMULATION SETUP FILES

## A.1 MOVING MESH

## Frame 3 – GDIidict dictionary

---

```

1 FoamFile
2 {
3     version      2.0;
4     format      ascii;
5     class       dictionary;
6     location     "constant";
7     object      GDIidict;
8 }
9 // * * * * * //
10 // Number of Valves
11     nValves 0;
12 //min Lift, below valve is considered as closed!
13     minLift 0.001; //0.0005;
14
15     piston_PatchName piston;
16
17     Piston
18     {
19         type cartesian;
20         origin (0 0 0);
21         axis (0 0 1);
22         direction (0 1 0);
23     }
24
25     //Valve Intake 1
26     //Intake valve name
27     valve1_BottomPatchName valveBottomIntake1;
28     valve1_StemPatchName valveStemIntake1;
29     valve1_TopPatchName valveTopIntake1;
30
31     //Valve Intake 2
32     //Intake valve name
33     valve2_BottomPatchName valveBottomIntake2;
34     valve2_StemPatchName valveStemIntake2;
35     valve2_TopPatchName valveTopIntake2;
36
37     //Valve Exhaust 3
38     //Intake valve name
39     valve3_BottomPatchName valveBottomExhaust1;
40     valve3_StemPatchName valveStemExhaust1;
41     valve3_TopPatchName valveTopExhaust1;

```

```

42
43 // Valve Exhaust 3
44 // Intake valve name
45 valve4_BottomPatchName  valveBottomExhaust2;
46 valve4_StemPatchName    valveStemExhaust2;
47 valve4_TopPatchName     valveTopExhaust2;
48
49 // lift profiles
50 liftProfileFile1  valve.lift.intake;
51 liftProfileFile2  valve.lift.intake;
52 liftProfileFile3  valve.lift.exhaust;
53 liftProfileFile4  valve.lift.exhaust;
54
55 // moving reference for the valve
56 Valve1_CoordinateSystem
57 {
58     type cylindrical;
59     origin ( -0.019 -0.0182 0.0059 );
60     axis (0 -0.390731 0.920505);
61     direction (1 0 0);
62 }
63 Valve2_CoordinateSystem
64 {
65     type cylindrical;
66     origin ( 0.019 -0.0182 0.0059 );
67     axis (0 -0.390731 0.920505);
68     direction (1 0 0);
69 }
70 Valve3_CoordinateSystem
71 {
72     type cylindrical;
73     origin ( -0.019 0.0182 0.0059 );
74     axis (0 0.390731 0.920505); // Y
75     direction (0 -0.920505 0.390731); // YY
76 }
77 Valve4_CoordinateSystem
78 {
79     type cylindrical;
80     origin ( 0.019 0.0182 0.0059 );
81     axis (0 0.390731 0.920505); // Y
82     direction (0 -0.920505 0.390731); // YY
83 }
84
85 // moving reference for the piston
86 Piston_CoordinateSystem
87 {
88     coordinateSystem
89     {
90     type cartesian;

```

```

91   origin (0 0 0);
92   coordinateRotation
93   {
94       type      axesRotation;
95       e1        (0 0 1);
96       e2        (0 1 0);
97       e3        (0 0 1);
98   }
99
100  }
101 }
102 ValveAxis
103 (
104   // Intake
105   (0 -0.390731 0.920505)
106   (0 -0.390731 0.920505)
107   // Exhaust
108   (0 0.390731 0.920505)
109   (0 0.390731 0.920505)
110 );

```

---

Source: OpenFOAM.

#### Frame 4 – engineGeometry dictionary

---

```

1 FoamFile
2 {
3   version      2.0;
4   format       ascii;
5   class        dictionary;
6   object       engineGeometry;
7 }
8 // * * * * * //
9
10 conRodLength  conRodLength [0 1 0 0 0 0 0] 0.148;
11 bore          bore [0 1 0 0 0 0 0] 0.086;
12 stroke        stroke [0 1 0 0 0 0 0] 0.086;
13 clearance     clearance [0 1 0 0 0 0 0] 0;
14 rpm           rpm [0 0 -1 0 0 0 0] 800;
15
16 checkMesh off;
17
18 engineTopoChangerMesh GDIFvMesh;
19
20 linerCentre (0 0 0);
21 linerAxis (0 0 1);
22 linerName liner;

```

---

Source: OpenFOAM.

---

 Frame 5 – dynamicMeshDict dictionary
 

---

```

1 FoamFile
2 {
3     version      2.0;
4     format      ascii;
5     class       dictionary;
6     location     "constant";
7     object      dynamicMeshDict;
8 }
9 // * * * * * //
10
11 //– Select the type of dynamicFvMesh
12 dynamicFvMesh      GDIFvMesh;
13 motionSolverLibs  ("libfvMotionSolvers.so");
14
15 //– Select the type of motionSolver
16 solver             velocityLaplacian;
17 // diffusivity     quadratic inverseDistance() //( valve1_intake_bottom
18     cylinderHead);
19 // diffusivity     directional (0 0 1);
20 // diffusivity     uniform;
21 // distancePatches (cylinderHead ValveBottom_1 ValveBottom_2 ValveBottom_3
22     ValveBottom_4);
23 // frozenDiffusion off;
24
25 velocityLaplacianCoeffs
26 {
27     diffusivity     uniform; // quadratic inverseDistance (cylinderHead
28     ValveBottom_1 ValveBottom_2 ValveBottom_3 ValveBottom_4 ) ;//(
29     valve1_intake_bottom cylinderHead);
30 distancePatches  (cylinderHead valveBottomIntake valveBottomExhaust);
31 }
32
33 GDIFvMeshCoeffs
34 {
35 }

```

---

Source: OpenFOAM.

## A.2 FUEL-SPRAY CLOUD

---

 Frame 6 – sprayCloudProperties dictionary
 

---

```

1 FoamFile
2 {
3     version      2.0;
4     format      binary;

```

```

5     class      dictionary;
6     location   "constant";
7     object     SprayCloudProperties;
8 }
9 // * * * * * //
10 solution
11 {
12     active      true;
13     coupled     true;
14     transient   yes;
15     cellValueSourceCorrection on;
16     maxCo       0.3;
17     sourceTerms
18     {
19         schemes
20         {
21             rho      explicit 1;
22             U         explicit 1;
23             Yi        explicit 1;
24             h         explicit 1;
25             radiation explicit 1;
26         }
27     }
28     interpolationSchemes
29     {
30         rho          cell;
31         U             cellPoint;
32         thermo:mu    cell;
33         T             cell;
34         Cp            cell;
35         kappa         cell;
36         p             cell;
37     }
38     integrationSchemes
39     {
40         U             Euler;
41         T             analytical;
42     }
43 }
44 constantProperties
45 {
46     T0               363;
47     rho0              690;
48     Cp0               4520;
49     constantVolume   false;
50 }
51 subModels
52 {
53     particleForces

```

```

54  {
55      sphereDrag ;
56  }
57
58  injectionModels
59  {
60      hollowCone
61      {
62          type          coneNozzleInjection ;
63          SOI           -76;
64          massTotal     2.9e-6; //13e-6
65          parcelBasisType mass ;
66          injectionMethod disc ;
67          flowType      pressureDrivenVelocity ;
68          Pinj          constant 18e6;
69          outerDiameter 0.165e-3;
70          innerDiameter 0;
71          duration      1.92;
72          position      ( 0 -0.0052 0.0103 );
73          direction     ( 0 1.5851 -10.7639 );
74          parcelsPerSecond 2e7;
75          flowRateProfile table
76          (
77              (0 1)
78              (0.0004 1)
79              (1000 0)
80          );
81
82          Cd            constant 0.95;
83          thetaInner    constant 53.5;
84          thetaOuter    constant 51.5;
85
86          sizeDistribution
87          {
88              type          RosinRammler ;
89              RosinRammlerDistribution
90              {
91                  minValue      1e-06;
92                  maxValue      100e-06;
93                  d              15e-6;
94                  n              3;
95              }
96          }
97      }
98  }
99
100 dispersionModel none ;
101 patchInteractionModel standardWallInteraction ;
102 heatTransferModel RanzMarshall ;

```

```

103 compositionModel singlePhaseMixture;
104 phaseChangeModel liquidEvaporationBoil;
105 surfaceFilmModel none;
106 atomizationModel none;
107 breakupModel ReitzDiwakar ; // ReitzKHRT;
108 stochasticCollisionModel none;
109 radiation off;
110 standardWallInteractionCoeffs
111 {
112     type rebound;
113 }
114 RanzMarshallCoeffs
115 {
116     BirdCorrection true;
117 }
118 singlePhaseMixtureCoeffs
119 {
120     phases
121     (
122         liquid
123         {
124             C8H18 1; //C7H16 1;
125         }
126     );
127 }
128 liquidEvaporationBoilCoeffs
129 {
130     enthalpyTransfer enthalpyDifference;
131     // activeLiquids ( C7H16 );
132     activeLiquids ( C8H18 );
133 }
134 ReitzDiwakarCoeffs
135 {
136     solveOscillationEq yes;
137     Cbag 6;
138     Cb 0.785;
139     Cstrip 0.5;
140     Cs 10;
141 }
142 ReitzKHRTCoeffs
143 {
144     solveOscillationEq yes;
145     B0 0.61;
146     B1 40;
147     Ctau 1;
148     CRT 0.1;
149     msLimit 0.2;
150     WeberLimit 6;
151 }

```



```

152 TABCoeffs
153 {
154     y0          0;
155     yDot0       0;
156     Cmu         10;
157     Comega      8;
158     WeCrit      12;
159 }
160 }

```

---

Source: OpenFOAM.

### A.3 COMBUSTION

#### Frame 7 – chemistryProperties dictionary

---

```

1 FoamFile
2 {
3     version      2.0;
4     format      ascii;
5     class       dictionary;
6     location     "constant";
7     object      chemistryProperties;
8 }
9 // * * * * * //
10
11 chemistryType
12 {
13     chemistrySolver  ode;
14     chemistryThermo  psi;
15 }
16
17 chemistry          on;
18
19 initialChemicalTimeStep 1e-07;
20
21 odeCoeffs
22 {
23     solver          seulex;
24     eps             0.05;
25 }

```

---

Source: OpenFOAM.

#### Frame 8 – combustionProperties dictionary

---

```

1 FoamFile
2 {
3     version      2.0;
4     format      ascii;

```

```

5     class      dictionary;
6     location   "constant";
7     object     combustionProperties;
8 }
9 // * * * * * //
10
11 combustionModel    laminar<psiChemistryCombustion>;
12 active             true;
13 laminarCoeffs
14 {
15 }
16
17 ignite             yes;
18
19 Tig Tig [0 0 0 1 0 0 0] 2500;
20
21 ignitionSites
22 (
23   {
24     location        (0 0.007175 0.005939);
25     diameter         0.002;
26     start            -19;
27     duration         10;
28     strength         4;
29   }
30 );
31
32 ignitionSphereFraction 1;
33 ignitionThickness ignitionThickness [0 1 0 0 0 0] 0;
34 ignitionCircleFraction 1;
35 ignitionKernelArea ignitionKernelArea [0 2 0 0 0 0] 0;

```

---

Source: OpenFOAM.

---

Frame 9 – CHEMKIN mechanism file for the single-step reaction of iso-octane/n-heptane

---

```

1 ELEMENTS
2 H O C N AR
3 END
4 SPECIE
5 C7H16 C8H18 O2 N2 CO2 H2O
6 END
7 REACTIONS
8 C8H18 + 12.5O2          => 8CO2 + 9H2O          4.60E+11 0.0 30000.0! 1
9     FORD / C8H18 0.25 /
10    FORD / O2 1.5 /
11 C7H16 + 11O2           => 7CO2 + 8H2O          5.00E+8 0.0 15780.0! 1
12    FORD / C7H16 0.25 /
13    FORD / O2 1.5 /
14 END

```

---

Source: OpenFOAM.

Frame 10 – NASA polynomials of species involved in the single-step reaction mechanism of iso-octane/n-heptane

---

```

1 THERMO ALL
2   200.000  1000.000  6000.000
3 C7H16          P10/85C  7.H 16.   0.   0.G  200.000  6000.000 1000.   1
4 2.04565203E+01 3.48575357E-02 -1.09226846E-05 1.67201776E-09 -9.81024850E-14 2
5 -3.25556365E+04 -8.04405017E+01 1.11532994E+01 -9.49419773E-03 1.95572075E-04 3
6 -2.49753662E-07 9.84877715E-11 -2.67688904E+04 -1.59096837E+01 -2.25846141E+04 4
7 C8H18          P 4/85C  8.H 18.   0.   0.G  200.000  6000.000 1000.   1
8 1.76160941E+01 5.13323108E-02 -1.65307266E-05 2.43232275E-09 -1.35572757E-13 2
9 -3.63461118E+04 -6.86446285E+01 8.15741071E-01 7.32647307E-02 1.78301503E-05 3
10 -6.93592790E-08 3.21630852E-11 -3.04774255E+04 2.41511097E+01 -2.69420567E+04 4
11 O2            ATcT06O  2.   0.   0.   0.G  200.000  6000.000 1000.   1
12 3.45852381E+00 1.04045351E-03 -2.79664041E-07 3.11439672E-11 -8.55656058E-16 2
13 1.02229063E+04 4.15264119E+00 3.78535371E+00 -3.21928540E-03 1.12323443E-05 3
14 -1.17254068E-08 4.17659585E-12 1.02922572E+04 3.27320239E+00 1.13558105E+04 4
15 N2            G 8/02N  2.   0.   0.   0.G  200.000  6000.000 1000.   1
16 2.95257637E+00 1.39690040E-03 -4.92631603E-07 7.86010195E-11 -4.60755204E-15 2
17 -9.23948688E+02 5.87188762E+00 3.53100528E+00 -1.23660988E-04 -5.02999433E-07 3
18 2.43530612E-09 -1.40881235E-12 -1.04697628E+03 2.96747038E+00 0.00000000E+00 4
19 CO2           L 7/88C  1O   2   0   0G  200.000  6000.000 1000.   1
20 0.46365111E+01 0.27414569E-02 -0.99589759E-06 0.16038666E-09 -0.91619857E-14 2
21 -0.49024904E+05 -0.19348955E+01 0.23568130E+01 0.89841299E-02 -0.71220632E-05 3
22 0.24573008E-08 -0.14288548E-12 -0.48371971E+05 0.99009035E+01 -0.47328105E+05 4
23 H2O           L 5/89H  2O   1   0   0G  200.000  6000.000 1000.   1
24 0.26770389E+01 0.29731816E-02 -0.77376889E-06 0.94433514E-10 -0.42689991E-14 2
25 -0.29885894E+05 0.68825500E+01 0.41986352E+01 -0.20364017E-02 0.65203416E-05 3
26 -0.54879269E-08 0.17719680E-11 -0.30293726E+05 -0.84900901E+00 -0.29084817E+05 4
27 END

```

---

Source: OpenFOAM.

Frame 11 – Reduced PRF reaction mechanism of Zhou et al. (2016)

---

```

1 ELEMENTS
2 H O N C AR
3 END
4 SPECIES
5 O2 CO2 H2O CO H2 OH H2O2
6 HO2 H O CH4 CH3O CH2O HCO CH3
7 C2H3 C2H4 C2H5 C3H4 C3H5 C3H6 C3H7 C4H9
8 C7H16 C7H15 C7H15O2 C7H14OOH O2C7H14OOH C7KET C5H11CO
9 C8H18 C8H17 C8H17O2 C8H16OOH O2C8H16OOH C8KET C6H13CO CH2OH
10 CH3OH C2H2 C3H3 HCCO CH2CO N2
11 END
12 REACTIONS
13 !*****n-heptAne oxidizAtion*****
14 ! *****FRom TsuRushimA*****

```

```

15 C7H16 +O2=C7H15+HO2          2.10E+16   0.00  4.600E+04
16 REV/ 1.000E+12 0.00 0.000E+00 /
17 C7H15 + O2=C7H15O2          2.20E+12   0.00  0.000E+00
18 REV/ 2.510E+13 0.00 2.740E+04 /
19 C7H15O2=C7H14OOH          2.20E+11   0.00  1.900E+04
20 REV/ 1.000E+11 0.00 1.100E+04 /
21 C7H14OOH + O2=O2C7H14OOH    3.16E+11   0.00  0.000E+00
22 REV/ 2.510E+13 0.00 2.740E+04 /
23 O2C7H14OOH=>C7KET + OH      1.25E+10   0.00  1.700E+04
24 C7KET=>C5H11CO + CH2O + OH   9.98E+15   0.00  4.300E+04
25 C7H16 + OH=>C7H15+H2O        0.05E+14   0.00  3.000E+03
26 C7H16+HO2=C7H15+H2O2        0.50E+13   0.00  1.695E+04
27 C5H11CO = C2H4 + C3H7 + CO   9.84E+15   0.00  4.020E+04
28 C7H15=> C2H5 + C2H4 + C3H6   1.05E+14   0.00  3.460E+04
29 C7H16 + H = C7H15 + H2      2.19E+10  2.00  4.760E+03
30 !
31 !*****iso-octAne oxidizAtion*****
32 ! *****FRom TsuRushimA*****
33 C8H18 +O2=C8H17+HO2          8.50E+16   0.00  4.600E+04
34 REV/ 1.000E+12 0.00 0.000E+00 /
35 C8H17 +O2=C8H17O2          2.10E+12   0.00  0.000E+00
36 REV/ 2.510E+13 0.00 2.740E+04 /
37 C8H17O2=C8H16OOH          2.10E+11   0.00  2.180E+04
38 REV/ 1.000E+11 0.00 1.100E+04 /
39 C8H16OOH + O2=O2C8H16OOH    2.58E+11   0.00  0.000E+00
40 REV/ 2.510E+13 0.00 2.740E+04 /
41 O2C8H16OOH=>C8KET + OH      3.90E+10   0.00  1.700E+04
42 C8KET=>C6H13CO + CH2O + OH   12.58E+15  0.00  4.300E+04
43 C6H13CO=>C4H9+C2H4+CO        2.84E+16   0.00  4.020E+04
44 C4H9 => C3H6 + CH3          18.48E+13  0.00  3.690E+04
45 C8H18 + OH=>C8H17+H2O        1.50E+12   0.00  3.000E+03
46 C8H18+HO2= C8H17+H2O2        0.50E+13   0.00  1.695E+04
47 C8H17=> C3H7 + C3H6 + C2H4   9.434E+17 -1.27 2.970E+04
48 C8H18 + H = C8H17 + H2      4.686E+10  2.00  7760
49 !*****FRom Wang*****
50 !
51 !
52 C3H7 = C2H4 + CH3            4.400E+13  0.00  30950.0
53 C3H7 = C3H6 + H              4.300E+14  0.00  36900.0
54 !*****C3H6 FROM UCSD_C4 MECHANISM
55 !*****C3H4*****
56 !*****C3H3*****
57 C3H4+OH=C3H3+H2O          5.3E6 2.0 2.0E3
58 C3H3+H(+M)=C3H4(+M)       3.0E13 0.0 0.0E0
59     LOW/9.0E15 1.0E0 0.0E0/
60     TROE/5.0E-1 1.0E30 1E-37/
61 C3H3+HO2=C3H4+O2          2.5E12 0.0 0.0E0
62 C3H3+O2=CH2CO+HCO         3.0E10 0.0 2.86807E3
63 C3H3+HCO=C3H4+CO          2.5E13 0.0 0.0E0

```

```

64 C3H3+HO2=OH+CO+C2H3      8.0E11 0.0 0.0E0
65 !*****C2H6*****
66 !*****C2H5*****
67 !*****C2H2*****
68 !*****
69 !C3H7          = C2H4      + CH3          9.600E+13 0.00 30950.0
70 !C3H7          = C3H6      + H           1.250E+14 0.00 36900.0
71 C3H6          = C2H3      + CH3          3.150E+15 0.00 85500.0
72 C3H6      + CH3 = C3H5      + CH4          9.000E+12 0.00 8480.0
73 C3H5      + O2  = C3H4      + HO2          6.000E+11 0.00 10000.0
74 C3H4      + OH  = C2H3      + CH2O         1.000E+12 0.00 0.0
75 C3H4      + OH  = C2H4      + HCO          1.000E+12 0.00 0.0
76 C2H5      + O2  = C2H4      + HO2          2.000E+10 0.0  -2200.0
77 C2H4      + OH  = CH2O      + CH3          6.000E+13 0.00 960.0
78 C2H4      + OH  = C2H3      + H2O          8.020E+13 0.00 5955.0
79 C2H3      + O2  = CH2O      + HCO          4.000E+12 0.00 -250.0
80 C2H3      + HCO = C2H4      + CO           6.034E+13 0.00 0.0
81 ! C2H2 ReActions
82 C3H5      = C2H2+CH3      2.397E+48 -9.90 8.208E+04
83 REV                          / 2.610E+46 -9.82 3.695E+04 /
84 C2H4(+M)  = C2H2+H2(+M)  1.800E+13 0.00 7.600E+04
85 LOW                          / 1.500E+15 0.00 5.544E+04 /
86 C2H3+O2   = C2H2+HO2     2.120E-06 6.00 9.484E+03
87 REV                          / 1.114E-07 6.33 1.757E+04 /
88 C2H3+H    = C2H2+H2     2.000E+13 0.00 2.500E+03
89 REV                          / 1.331E+13 0.00 6.808E+04 /
90 C2H2+H(+M) = C2H3(+M)    3.110E+11 0.58 2.589E+03
91 LOW                          / 2.254E+40 -7.269 6577. /
92 TROE/ 1.0 1.E-15 675. 1.E+15/
93 H2/2/ H2O/5/ CO/2/ CO2/3/
94 C2H2+O2   = HCCO+OH      2.000E+08 1.50 3.010E+04
95 REV                          / 2.232E+05 1.50 2.540E+04 /
96 C2H2+O    = HCCO+H        1.430E+07 2.00 1.900E+03
97 REV                          / 2.021E+05 2.00 1.331E+04 /
98 C2H2+OH   = CH2CO+H      2.190E-04 4.50 -1.000E+03
99 REV                          / 2.161E-03 4.50 1.966E+04 /
100 CH2CO+H  = CH3+CO        1.100E+13 0.00 3.400E+03
101 REV                          / 2.400E+12 0.00 4.020E+04 /
102 CH2CO+O  = HCCO+OH      1.000E+13 0.00 8.000E+03
103 REV                          / 1.432E+10 0.00 -1.255E+03 /
104 CH2CO+OH = HCCO+H2O      1.000E+13 0.00 2.000E+03
105 REV                          / 1.412E+11 0.00 9.995E+03 /
106 CH2CO+H  = HCCO+H2      2.000E+14 0.00 8.000E+03
107 REV                          / 6.522E+11 0.00 8.400E+02 /
108 HCCO+OH  = HCO+HCO       1.000E+13 0.00 0.000E+00
109 REV                          / 2.411E+14 0.00 4.036E+04 /
110 HCCO+O   = H+CO+CO       8.000E+13 0.00 0.000E+00
111 REV                          / 0.000E+00 0.00 0.000E+00 /
112 HCCO+O2  = CO2+HCO       2.400E+11 0.00 -8.540E+02

```

```

113 REV / 1.474E+14 0.00 1.336E+05 /
114 !*****Ch3OH
115 ! ***** H2-O2 ChAin ReActions *****
116 H+O2=O+OH 3.547e+15 -0.406 1.6599E+4
117 O+H2=H+OH 0.508E+05 2.67 0.629E+04
118 H2+OH=H2O+H 0.216E+09 1.51 0.343E+04
119 O+H2O=OH+OH 2.97e+06 2.02 1.34e+4
120 ! ***** H2-O2 DissociAtion ReActions *****
121 H2+M=H+H+M 4.577E+19 -1.40 1.0438E+05
122 H2/2.5/ H2O/12/
123 CO/1.9/ CO2/3.8/
124 O+O+M=O2+M 6.165E+15 -0.50 0.000E+00
125 H2/2.5/ H2O/12/
126 CO/1.9/ CO2/3.8/
127 O+H+M=OH+M 4.714E+18 -1.00 0.000E+00
128 H2/2.5/ H2O/12/
129 CO/1.9/ CO2/3.8/
130 H+OH+M=H2O+M 3.800E+22 -2.00 0.000E+00
131 H2/2.5/ H2O/12/
132 CO/1.9/ CO2/3.8/
133 !***** FoRmAtion And Consumption of HO2*****
134 H+O2(+M)=HO2(+M) 1.475E+12 0.60 0.00E+00
135 LOW/6.366E+20 -1.72 5.248E+02/
136 TROE/0.8 1E-30 1E+30/
137 H2/2.0/ H2O/11./ O2/0.78/ CO/1.9/ CO2/3.8/
138 HO2+H=H2+O2 1.66E+13 0.00 0.823E+03
139 HO2+H=OH+OH 7.079E+13 0.00 2.95E+02
140 HO2+O=O2+OH 0.325E+14 0.00 0.00E+00
141 HO2+OH=H2O+O2 2.890E+13 0.00 -4.970E+02
142 ! ***** FoRmAtion And Consumption of H2O2*****
143 HO2+HO2=H2O2+O2 4.200e+14 0.00 1.1982e+04
144 DUPLICATE
145 HO2+HO2=H2O2+O2 1.300e+11 0.00 -1.6293e+3
146 DUPLICATE
147 H2O2(+M)=OH+OH(+M) 2.951e+14 0.00 4.843E+04
148 LOW/1.202E+17 0.00 4.55E+04/
149 TROE/0.5 1E-30 1E+30/
150 H2/2.5/ H2O/12/
151 CO/1.9/ CO2/3.8/
152 H2O2+H=H2O+OH 0.241E+14 0.00 0.397E+04
153 H2O2+H=HO2+H2 0.482E+14 0.00 0.795E+04
154 H2O2+O=OH+HO2 9.550E+06 2.00 3.970E+03
155 H2O2+OH=HO2+H2O 1.000E+12 0.00 0.000
156 DUPLICATE
157 H2O2+OH=HO2+H2O 5.800E+14 0.00 9.557E+03
158 DUPLICATE
159 !***** CO/HCO REACTIONS *****
160 CO+O(+M)=CO2(+M) 1.80E+10 0.00 2384.
161 LOW/1.55E+24 -2.79 4191./

```

```

162      H2/2.5/ H2O/12/ CO/1.9/ CO2/3.8/
163 CO+O2=CO2+O          0.253E+13  0.00  0.477E+05
164 CO+HO2=CO2+OH        3.01E+13   0.00  2.30E+04
165 CO+OH=CO2+H          2.229E+05  1.89  -1158.7
166 HCO+M=H+CO+M         4.7485E+11 0.659 1.4874E+04
167 H2/2.5/ H2O/6/ CO/1.9/ CO2/3.8/
168 HCO+O2=CO+HO2        0.758E+13  0.00  0.410E+03
169 HCO+H=CO+H2          0.723E+14  0.00  0.000E+00
170 HCO+O=CO+OH          0.302E+14  0.00  0.000E+00
171 HCO+OH=CO+H2O        0.302E+14  0.00  0.000E+00
172 HCO + O = CO2 + H          3.000E+13  0.00  0.000E+00
173 HCO + HO2 = CO2 + OH + H   3.000E+13  0.00  0.000E+00
174 HCO + CH3 = CO + CH4       1.200E+14  0.00  0.000E+00
175 HCO + HCO = H2 + CO + CO   3.000E+12  0.00  0.000E+00
176 HCO + HCO = CH2O + CO      3.0E+13   0.00  0.00
177 !***** ch2o ReActions *****
178 CH2O + M = HCO + H + M       3.3E+39   -6.3  9.99E+04
179 H2/2.5/ H2O/12.0/ CO/1.9/ CO2/3.8/
180 CH2O + M = CO + H2 + M       3.1E+45   -8.0  9.751E+04
181 H2/2.5/ H2O/12.0/ CO/1.9/ CO2/3.8/
182 CH2O + H = HCO + H2          5.74E+07  1.9  2.7486E+03
183 CH2O + O = HCO + OH          1.810E+13  0.00  3.080E+03
184 CH2O + OH = HCO + H2O        3.430E+09  1.18 -4.470E+02
185 CH2O + O2 = HCO + HO2        1.23e+6   3.00  52000.
186 CH2O + HO2 = HCO + H2O2      4.11E+4   2.5  10210.
187 CH2O+CH3 = HCO+CH4  3.636E-06  5.42  9.980E+02
188 !***** ch4 ReActions *****
189 CH3 + O = CH2O + H            8.430E+13  0.00  0.000E+00
190 CH3 + O2 = CH3O + O           1.990E+18 -1.57  2.923E+04
191 CH3 + O2 = CH2O + OH          3.74e+11  0.0  14640
192 CH3 + HO2 = CH3O + OH         2.41e+10  0.76 -2325
193 CH3+H(+M) = CH4(+M)          1.270E+16 -0.630 383.00
194      LOW / 2.477E+33 -4.760 2440.00/
195      TROE/ 0.7830 74.00 2941.00 6964.00 /
196      H2/2.0/ H2O/6.0/ CH4/2.0/ CO/1.5/ CO2/2.0/
197 CH4 + H = CH3 + H2           5.470E+07  1.97  1.121E+04
198 CH4 + O = CH3 + OH           3.150E+12  0.50  10290.0
199 CH4 + OH = CH3 + H2O         5.720E+06  1.96  2.639E+03
200 CH3 + HO2 = CH4 + O2          3.16E+12  0.00  0.00!3.16
201 CH4 + HO2 = CH3 + H2O2       1.810E+11  0.00  1.858E+04
202 !***** ch3oh/ch2oh ReActions *****
203 CH2OH + M = CH2O + H + M     1.0E+14  0.0  25.1E+3
204 CH2OH + H = CH2O + H2        6.000E+12  0.00  0.000E+00
205 CH2OH + H = CH3 + OH         9.635E+13  0.00  0.000E+00
206 CH2OH + O = CH2O + OH        4.200E+13  0.00  0.000E+00
207 CH2OH + OH = CH2O + H2O      2.400E+13  0.00  0.000E+00
208 CH2OH + O2 = CH2O + HO2      2.410E+14  0.00  5.017E+03
209      DUP
210 CH2OH + O2 = CH2O + HO2      1.510E+15 -1.00  0.000E+00

```

```

211     DUP
212 CH2OH + HO2 = CH2O + H2O2          1.200E+13  0.00  0.000E+00
213 CH2OH + HCO = CH3OH + CO           1.000E+13  0.00  0.000E+00
214 CH2OH + HCO = CH2O + CH2O          1.500E+13  0.00  0.000E+00
215 !***** Ethylene glycol foRmAtion *****
216 2CH2OH = CH3OH + CH2O               3.000E+12  0.00  0.000E+00
217 CH2OH + CH3O = CH3OH + CH2O        2.400E+13  0.00  0.000E+00
218 !***** CH3O ReActions *****
219 CH3O + M = CH2O + H + M              8.300E+17 -1.20  1.550E+04 !MC Lin
220 CH3O + H = CH3 + OH                 3.200E+13  0.00  0.000E+00
221 CH3O + O = CH2O + OH                6.000E+12  0.00  0.000E+00
222 CH3O + OH = CH2O + H2O              1.800E+13  0.00  0.000E+00
223 CH3O + O2 = CH2O + HO2              9.033E+13  0.00  1.198E+04
224     DUP
225 CH3O + O2 = CH2O + HO2              2.200E+10  0.00  1.748E+03
226     DUP
227 CH3O + HO2 = CH2O + H2O2            3.000E+11  0.00  0.000E+00
228 CH3O + CO = CH3 + CO2               1.600E+13  0.00  1.180E+04
229 CH3O + HCO = CH3OH + CO             9.000E+13  0.00  0.000E+00
230 2CH3O = CH3OH + CH2O               6.000E+13  0.00  0.000E+00
231 !***** CH3OH ReActions *****
232 OH+CH3(+M)<=>CH3OH(+M)                2.790E+18   -1.430   1330.00
233     LOW / 4.000E+36   -5.920   3140.00/
234     TROE/ .4120  195.0  5900.00  6394.00/
235 H2/2.00/ H2O/6.00/ CH4/2.00/ CO/1.50/ CO2/2.00/
236 H+CH2OH(+M)<=>CH3OH(+M)                1.055E+12   .500   86.00
237     LOW / 4.360E+31   -4.650   5080.00/
238     TROE/ .600  100.00  90000.0  10000.0 /
239 H2/2.00/ H2O/6.00/ CH4/2.00/ CO/1.50/ CO2/2.00/
240 H+CH3O(+M)<=>CH3OH(+M)                2.430E+12   .515   50.00
241     LOW / 4.660E+41   -7.440   14080.0/
242     TROE/ .700  100.00  90000.0  10000.00 /
243 H2/2.00/ H2O/6.00/ CH4/2.00/ CO/1.50/ CO2/2.00/
244 CH3OH + H = CH2OH + H2              3.200E+13  0.00  6.095E+03
245 CH3OH + H = CH3O + H2               8.000E+12  0.00  6.095E+03
246 CH3OH + O = CH2OH + OH              3.880E+05  2.50  3.080E+03
247 CH3OH + OH = CH3O + H2O             1.000E+06  2.10  4.967E+02
248 CH3OH + OH = CH2OH + H2O            7.100E+06  1.80 -5.960E+02
249 CH3OH + O2 = CH2OH + HO2            2.050E+13  0.00  4.490E+04
250 CH3OH + HCO = CH2OH + CH2O          9.635E+03  2.90  1.311E+04
251 CH3OH + HO2 = CH2OH + H2O2          3.980E+13  0.00  1.940E+04
252 CH3OH + CH3 = CH2OH + CH4           3.190E+01  3.17  7.172E+03
253 CH3O + CH3OH = CH3OH + CH2OH        3.000E+11  0.00  4.060E+03
254 END

```

---

Source: Zhou et al. (2016).