KARLSRUHE INSTITUTE OF TECHNOLOGY
Steinbuch Centre for Computing (SCC)

Master Thesis on

Study of the Modelling of Two Phase Flows for Diesel Engine Combustion

Author: *Matteo Davide Villani* (ID: 2233892)
German Supervisor: *Prof. Dr.-Ing. Thomas Koch*
Assistants: *Dr.-Ing. Jordan Denev, Dipl.-Ing. Georg Blesinger*
Italian Supervisor: *Prof. Dr.-Ing. Daniela Anna Misul*

Department of Mechanical Engineering

Politecnico di Torino

October, 2019

Declaration of Authenticity

I declare that I have written the work myself, that I have specified all the tools and sources used completely and accurately, that I have specified everything that has been taken from the work of others, either unchanged or with modifications, and that I have payed attention to the statutes of KIT for the safeguarding of good scientific practice as amended from time to time.

Signature:

Date:

Acknowledgment

I would like to thank my parents first of all for their support during these university years and for never letting me feel under pressure.

I want to thank all the friends who have been close to me over the years, those I have known and those I already knew.

Thanks to the Karlsruhe Institute of Technology and the Politecnico di Torino for giving me the opportunity to study and to do an exchange experience like the Erasmus project.

And in the end, I would like to thank myself for the stubbornness, the desire to do, to know and to improve myself.

Abstract

The important improvements of the past decades have made necessary the use of Computation Fluid Dynamics (CFD) tools, especially for the regulations in force in terms of polluting emissions, but which must above all meet the needs imposed by technological development in other fields.

Mostly the study and the design of tools, that use the interaction between two phases, needs the use of CFD tools, due to the difficulties in direct experimental measurements of the optically thick liquid core in the liquid phase injection, which occurs at high temporal and spatial resolutions. The study of the spray injected into a gas phase is quite interesting, because allows to understan all the physics, the thermofluidynamics behind the interaction between a liquid phase injected (with particles studied and approximated according to different methodologies) and a gas phase (which can be interested by a laminar or a turbulent flow, with all the phenomena that come from it).

The present work studies a liquid phase injected into a rectrangular channel in non-reactive conditions, using OpenFOAM, a freeware CFD software. This work focuses on Eulerian-Lagrangian study of the interaction between the liquid particles, injected into the duct, and the air gas-flow within the duct. This work allows to focus the attention on how the initial and boundary conditions of the physical case influence the lifetime of certain phenomena (like evaporation) or other phsical quantities (like the liquid penetration length).

Among the different solvers available on OpenFOAM to study each physical case, our choice has been the solver sprayFoam, that gives useful simulation results for the study of a liquid spray. The case studied involves a $k - \epsilon$ turbulence model, and a $KHRT$ break-up scheme, which represents a specific model that describes how liquid droplets injected into a gas flow unite/divide giving rise to new droplets.

Table of contents

1 Introduction **1**

2 Objectives **2**

3 Theory and Modeling **3**

- 3.1 Background . . . 3
- 3.2 Governing Equations . . . 3
- 3.3 Eulerian Phase . . . 4
  - 3.3.1 Mass equation . . . 4
  - 3.3.2 Momentum Equation . . . 6
  - 3.3.3 Energy Equation . . . 7
  - 3.3.4 Turbulence Equation(RANS model) . . . 9
  - 3.3.5 Thermophysical models . . . 11
- 3.4 Lagrangian Phase . . . 12
  - 3.4.1 Mass Equation . . . 12
  - 3.4.2 Momentum Equation . . . 15
  - 3.4.3 Energy Equation . . . 15
- 3.5 Particle Forces . . . 16
  - 3.5.1 Drag force . . . 17
  - 3.5.2 Gravity/Buoyancy and Pressure Gradient Force . . . 17
  - 3.5.3 Other Forces . . . 18
- 3.6 Particle Response Time . . . 19
- 3.7 Phase-Coupling Mechanism . . . 20
- 3.8 Particle-Particle Interaction . . . 20
- 3.9 Sthocastic Dispersion Model . . . 21
  - 3.9.1 Turbulence effect on particle dispersions . . . 22
  - 3.9.2 Stochastic model . . . 22
  - 3.9.3 Discrete Random Walk . . . 23
- 3.10 Droplets Breakup . . . 25
  - 3.10.1 Primary breakup . . . 26
  - 3.10.2 Seconday breakup: the KHRT breakup model . . . 27
  - 3.10.3 Another secondary breakup model: ReitzDiwakar model 30
  - 3.10.4 Why didn't we use other secondary breakup models? . 31
- 3.11 Rosin-Rammler Distribution . . . 32
- 3.12 Particle size monitoring . . . 33
  - 3.12.1 Sauter Mean Diameter . . . 33

4 Numerical methods 35
4.1 OpenFOAM and SprayFoam 35
4.2 Relationship between the lagrangian study and the eulerian one 36
4.3 Parcel and Spray Concept 36
4.3.1 Gas-Liquid Coupling 37
4.3.2 Liquid-Gas Coupling 38
4.3.3 Pressure and Velocity Coupling 39
4.4 The Courant/CFL Number 40
4.5 Simulations on the bwUnicluster 41

5 CFD Analysis of the duct with one liquid droplet 43
5.1 The liquid 43
5.2 sprayCloudProperties file 43
5.3 Our model configuration 44
5.4 Comparison with other models 45
5.5 How evaporation time changes by varying the initial conditions 47
5.5.1 Changing particle speed 47
5.5.2 Changing particle size 49
5.5.3 Changing surrounding pressure 50
5.6 How evaporation time changes by varying the characteristics of the mesh 53
5.6.1 Changing the size of the Mesh in all directions 53
5.6.2 Changing the size of the Mesh in x-direction 55
5.6.3 Changing the size of the Mesh in y-direction and z-direction 57
5.6.4 Relative Errors and the best Mesh settings 59

6 CFD Analysis of the duct with the liquid spray jet 61
6.1 Our model configuration 61
6.2 Validation of the model 62
6.3 Penetration lengths 64
6.3.1 Procedure to measure the vapour penetration length . 66
6.4 Geometric field and mesh size 67
6.5 Phase change 68
6.6 Breakup models comparison 70
6.7 Turbulence effects comparison 74
6.7.1 Turbulence on/off 75
6.7.2 LES turbulence model 76

7 Conclusions 78
7.1 Future improvements 79

References **80**

1 Introduction

In the present work the main objective is to validate a fluid dynamics simulation model on OpenFOAM to study the interaction between the liquid phase (in the form of particles, injected as a spray) and the gas phase, through some parametric studies.

OpenFOAM is an open source program that is growing rapidly in recent years, expanding its user community. Each user can develop his own solver, like Lib-ICE [1] developed by Politecnico di Milano, in order to use the great power of this software and to simulate whatever physical case one needs.

The only problem, related to OpenFOAM, is the absence of literature , except for the User Guide [2] provided by the OpenFOAM Foundation; so one of our main goals was also to study the software and understand how the different modelling parts work; so that we can validate our own model, which can be the starting point for further studies.

Therefore the work develops with a preliminary study of a single drop injected into a channel. The study of single drop injection is useful for validating our model, comparing the data obtained from a similar CFD simulation carried out by Likun Ma, and Drik Roekaerts [3], and also referring to the analytical data obtained from a Matlab Model taken from Andreas Höfler [4].

Then,the study focuses on the liquid spray itself and all the phenomena that take part of its injection and its interaction with the gas phase. From time to time, using experimental data or results from other simulations, provided to us by other researchers, we've seen how to improve our model, in order to make it reliable and realistic.

The case of the liquid spray obviously requires more attention and the use of different theories and different thermofluid dynamics models, to make the simulation as realistic as possible. With more than one particle, it's necessary to introduce the different atomization phenomena that take part to the injection of the spray.

For example, the break-up model, chosen among different break-up models, analyzed by S.Hossainoiyr, A.R. Binesh [5] and by P.Ghadimi, M.Yousefifard, H. Nowunzi [6]; with the final choice strengthened by referring to the study of Ghasemi [1], that of Alejandro Municio Aránguez [7].

2 Objectives

CFD software is a very effective means of studying very complex systems, which require the solution of complicated differential equations. Precisely for this reason our choice is OpenFOAM, which has many advantages: it is a very flexible system, being a C ++ library that can be modified by the user according to the needs, it has several solvers to be able to deal with various types of systems and, moreover, it's free.

For this reason, the first goal of this work is to study the main features of OpenFOAM and its capabilities with the solver that best fits the physical situation we want to describe. The solver used is sprayFoam.

Thanks to the solver sprayFoam, we were able to follow our second goal, namely to study and simulate liquid-gas multiphase flows (with the liquid phase intended as a spray) through the Eulerian-Lagrangian approach.

The use of the Langrangian Particle Tracking (LPT) method to study the liquid spray represents the most practical method and it also requires less computational effort compared to the Eulerian method (used to study the gas phase). Thanks to the LPT method we'll be able to analyze the most important phenomena, as atomization, transport and evaporation, through a parametric study to unserstand which are the most important initial conditions.

To achieve these two goals we can use some available test cases as test object to practice the modelling, to evaluate different ways of modelling, to identify modelling requirements (time steps, grid resolution) and modelling limitations (like accuracy or plausibility).

We'll start with a simple case (that of the single droplet) to understand how to model evaporization and transport phenomena for varying initial conditions (pressure, temperature, slip velocity, droplet diamater) and thermo-physical properties (related to substances).

Once the study of the simple case (that of the single droplet) is completed, we'll be able to switch to the liquid spray and to understand which study models (e.g. breakup models) and methods (e.g. particle size distribution methods) are the best to achieve plausible results and in agreement with what is obtained experimentally.

3 Theory and Modeling

3.1 Background

The study of a liquid spray injected into a certain domain is interesting because it allows to study how liquid particles behave when they interact with a gas phase, but also when they interaction with each other (particles can also collide, and then join or divide). It's interesting to see which are the forces that modify their path and how they change by mutating the initial condtions (i.e. the geometry, the temperature, the pressure, the density) and the boundary conditions of the physical case.

In the description of the interaction between two different phases like that described in this work, it's necessary to focus the attention on a lot of physical phenomena that are usually not considered or even known. The surrounding gas phase is studied thorugh the Eulerian approach, which allows us to define a static grid (the mesh) through which we will study how the physical and thermodynamic properties will change over time, depending on the interaction with the liquid particles or the initial conditions of the gas itself.

About the liquid particles, we use the Lagrangian Particle Tracking (LPT) method that, differently from the Eulerian method, is not a static set of points that studies how much the properties in the zone surrounded by them changes, but is instead a method that studies each particle, following it individually in space and time.

Once introduced how we'll study the spray evolution during the injection, we can introduce the governing equations of the Eulerian-Lagrangian methodology.

3.2 Governing Equations

To introduce anlytically our model we've to specify that the fluid mechanics uses three equations to describe the behavior of a flow, which are adopted both for the Eulerian and for the Lagrangian method, but with a different type of derivatives: it's possible to distinguish the Lagrangian derivative and the Eulerian one. So the three types of equations, that become become nine because of the three axial coordinates in the space (x,y,z), are showed into the next page:

- **Mass Equation:** which is related to mass evolution in time;
- **Momentum equations:** obtainable from the law of conservation of momentum in the three spatial directions;
- **Energy equation:** related to the conservation of energy.

They are usually known as the **"Navier-Stokes" equations**.

After introducing the three families of equations that are fundamental for the fluid-dynamic study of a flow, we can now explain and describe the Eulerian phase and the Lagrangian one.

3.3 Eulerian Phase

The Eulerian phase consists into the description of the continuous gas phase, surrounding the liquid particles of the spray, through a static mesh grid, that studies the development brought by the spray injected into the channel, that interacts with this gas phase. The grid is made of different points and throught the interpolation of the data collected from each point, is possible to study the fluid behaviour in a particular zone of the grid.

Thus mass, momentum and energy equation of the gas phase are solved by adopting the Reynolds-averaged Navier–Stokes (RANS) approach, that uses time-averaged equations of motion.

3.3.1 Mass equation

Applying the mass conservation for each infinitesimal volume of the static grid applied to the gas phase, we can get the continuity equation for the Eulerian phase, that is reported below:

$$\frac{\partial \overline{\rho}}{\partial t} + \frac{\partial}{\partial x_i}(\overline{\rho}\ \overline{u_i}) = \dot{S}_p \tag{1}$$

We've to underline three terms of the equation: $\overline{\rho}$, that indicates the time averaged gas density, $\overline{u_i}$ is the time averaged component i of the velocity vector field and $\dot{S}_p$ is a source term, that represents the mass of the spray injected. The source term $\dot{S}_p$ is important to guarantee the contribution in the conversation of the mass, with an increase or a descrease of it, by the Lagrangian phase to the Eulerian one.

Species mass fraction's conservation is described by:

$$\frac{\partial \overline{\rho} \overline{Y_n}}{\partial t} + \frac{\partial}{\partial x_i}(\overline{\rho}\ \overline{u_i Y_n}) + \frac{\partial}{\partial x_i}(\frac{\overline{J_n}}{x_i}) = \dot{S}_{p,n} + \dot{S}_{chem,n} \tag{2}$$

where $\overline{J_n}$ represents the time averaged mass diffusion flux, $\overline{Y_n}$ is the time averaged mass fraction of species n, $S_{p,in}$ is the rate of creation by addition from the dispersed phase plus any other sources (in our case the spray injected into the flow), and, finally, $S_{chem,n}$ is the net rate of production of species n by chemical reactions.

The mass diffusion flux J_n is defined as:

$$J_n = -\left(\rho D_{n,diff} + \frac{\mu_t}{S_{ct}}\right)\frac{\partial Y_n}{\partial x_i} \tag{3}$$

where $S_{ct} = \frac{\mu_t}{\rho D_t}$ is the *turbulent Schmidt number*, while $D_{n,diff}$ represents the *diffusion coefficent* for the species n in the mixture.

The source term $\dot{S}_p$ is defined as:

$$\dot{S}_p = \sum_{l=1}^{n}(\dot{m}_{lg} - \dot{m}_{gl}) \tag{4}$$

where $\dot{m}_{lg}$ and $\dot{m}_{gl}$ represent the mass transfer rate from the liquid phase to the gaseous one and viceversa, while n stands for the number of phases.

It is not necessary to define the term $\dot{S}_{chem,n}$, which introduces other coefficients that we will not address in this discussion, since we are considering non-reactive species.

3.3.2 Momentum Equation

Applying the Newton's second law, is possible to get the momentum equation, that has to be applied for each of the three spatial coordinates (x,y,z):

$$\frac{\partial(\overline{\rho}\ \overline{u_i})}{\partial t} + \frac{\partial}{\partial x_i}(\overline{u_j}\ \overline{\rho}\ \overline{u_i}) = -\frac{\partial \overline{p}}{\partial x_j} + \frac{\partial \overline{\sigma_{ij}}}{\partial x_i} + \frac{\partial \overline{\tau_{ij}}}{\partial x_i} + \dot{S_M} \tag{5}$$

For this equation we've to underline three terms: $\overline{p}$ represents the uniform pressure field in the cell considered, $\overline{\sigma_{ij}}$ is the viscous stress tensor, while $\overline{\tau_{ij}}$ is the averaged Reynolds stress tensor. $\dot{S}_M$, instead, is a source term related to lagrangian approach of spray simulation, and, as in analogy to the continuity equation, allows us to connect the eulerian phase to the lagrangian one, in order to not ignore the inertial contribution of the liquid particles, that influence the momentum of the gas phase.

The viscous stress tensor $\overline{\sigma_{ij}}$ is approximately equal to:

$$\overline{\sigma_{ij}} \approx 2\mu\left(\overline{S_{ij}} - \frac{1}{3}\frac{\partial \overline{u_k}}{\partial x_k}\delta_{ij}\right) \tag{6}$$

with the term $\overline{S_{ij}} = \frac{1}{2}(\frac{\partial \overline{u_j}}{\partial x_j} + \frac{\partial \overline{u_j}}{\partial x_i})$,

and the Kronecker delta δ_{ij} ($\delta_{ij} = 1$, for $i = j$, $\delta_{ij} = 0$, for $i \neq j$).

While, the Reynolds stress tensor τ_{ij} is approximately equal to:

$$\tau_{ij} = -\overline{\rho u_i' u_j'} = \mu_t\left(\frac{\partial \overline{u_i}}{\partial x_j} + \frac{\partial \overline{u_i}}{\partial x_i}\right) \tag{7}$$

with μ_t that represents the dynamic turbulent kinematic viscosity, explained in the Eq. 16. It is important to specify that this is the definition given by the RANS method, while using the LES method, we've also to subtract the term $\mu\frac{2}{3}\frac{\partial u_k}{x_k}S_{ij}$.

The source term $\dot{S_M}$, whose definition is taken from the *Ansys Manual* [8] and the *Siemens User Guide* [9], considers the liquid particles studied through the lagrangian approach and it's important to be defined, in order to underline which is the term that connects the eulerian approach and the lagrangian one:

$$\dot{S_M} = \sum_{l=1}^{n} (K_{gl}(U^{(l)} - U) + \dot{m}_{lg} u_{lg} - \dot{m}_{gl} u_{gl}) \tag{8}$$

where K_{gl} is the interphase momentum exchange coefficent, $U^{(l)}$ the velocity of the liquid phase, u_{gl} and u_{lg} the interphase velocties. About the summation, we've that it is referred to the number of phases considered during the calculation.

The *interphase momentum exchange coefficent* K_{gl}, part of the energy source term $\dot{S_M}$ is equal to:

$$K_{gl} = \frac{C_D Re}{8 d_{particle}} \mu_g A_i \tag{9}$$

where C_D is the drag coefficent, defined in the sub-section 3.5.1 of this work, Re is the Reynolds Number, $d_{particle}$ is the mean diameter of the particles, A_i is the surface of interaction between the gaseous and the liquid phase (the external surface of each particle) and, finally, μ_g is the dynamic viscosity of the gaseous phase surrounding the liquid droplet.

3.3.3 Energy Equation

The equation is only the result of the first law of thermodynamics: it is to consider the balance between how much between the speed with which the energy of each particle changes and the heat flow that is added to the particle and the work done on it.

The goal is to write the energy equation as function of the specific total enthalpy, so we need to define the specific total enthalpy, whose definition is taken from the *Anyis Manual* [8] : it's the sum of the specific static enthalpy, specific dynamic enthalpy and the contribute given by the turbulent kinetic energy k.

Thus, the *specific total enthalpy* is equal to:

$$h_{tot} = h + \frac{1}{2}u_i u_j + k \tag{10}$$

In which h is the *static enthalpy*, $\frac{1}{2}u_i u_j$ is the *dynamic enthalpy*, and k is the *turbulent kinetic energy*, that represents the contribution given by turbulence to the total enthalpy.

The turbulent kinetic energy is defined as:

$$k = \frac{1}{2}u_i'^2 \tag{11}$$

Thus, the energy equation as function of the specific total enthalpy, can be written as follows:

$$\begin{aligned} \frac{\partial\overline{\rho}\ \overline{h_{tot}}}{\partial t} - \frac{\partial\overline{p}}{\partial t} + \frac{\partial}{x_i}(\overline{\rho}\ \overline{u_i}\ \overline{h_{tot}}) = \frac{\partial}{\partial x_i}(\lambda\frac{\partial\overline{T}}{\partial x_i} - \overline{\rho}\overline{u_i'}\overline{h'} + \\ + \frac{\partial}{\partial x_i}[(\overline{u_j}(\overline{\sigma_{ij}} - \overline{\rho u_i' u_j'})] + \dot{S}_Q \end{aligned} \tag{12}$$

with λ that represents the *thermal conductivity.*

At the end, we define the source term $\dot{S}_Q$ that represents the contribution of the liquid particles injected into the duct, taking care also of the heat exchange intensity Q_{lg} between the liquid and the gaseous phase:

$$\dot{S}_Q = \sum_{l=1}^{n}(Q_{lg} + \dot{m}_{lg}h_{lg} - \dot{m}_{gl}h_{gl}) \tag{13}$$

in which h_{lg} and h_{gl} are the *specific interphase enthalpies*, and represent the heat required to produce one phase from the other. The heat Q_{lg} is the heat transferred by thermal diffusion between the gaseous phase and the liquid one; while mass is considered to be transferred only by phase change (evaporation or condensation), and it's computed thorugh a heat balance.

3.3.4 Turbulence Equation(RANS model)

The turbulence phenomenon is studied, as we've specified some sub-chapters ago, through the Reynolds-averaged Navier Stokes model of equations, that in our case is implemented through the standard $k-\epsilon$ model, that takes care of the turbulent kinetic energy k and of the dissipation rate ϵ to model the spray injection.

- The turbulent kinetic energy k is given by:

$$\begin{aligned}\frac{\partial}{\partial t}(\overline{\rho}\overline{k}) + \frac{\partial}{\partial x_i}(\overline{\rho}\overline{k}\overline{u_i}) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + \\ -\overline{\rho u_i' u_j'}\frac{\partial \overline{u_j}}{\partial x_i} + 2\mu_t S_{ij}S_{ij} - \overline{\rho}\epsilon - G_k + S_k\end{aligned} \tag{14}$$

- The dissipation rate ϵ is given by:

$$\begin{aligned}\frac{\partial}{\partial t}(\overline{\rho}\epsilon) + \frac{\partial}{\partial x_i}(\overline{\rho}\epsilon\overline{u_i}) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_j}\right] + \\ +C_1\frac{\epsilon}{k}(2C_3\mu_t S_{ij}S_{ij} - \overline{\rho u_i' u_j'}\frac{\partial \hat{u}_j}{\partial x_i}) - C_2\overline{\rho}\frac{\epsilon^2}{k} + S_\epsilon\end{aligned} \tag{15}$$

in these two equation we've different coefficents to specify: G_k is the mean rate of deformation tensor, C_1 is a constant that affects the mixing of the spray and its penetration lengths, σ_k and σ_ϵ are the turbulent Prandtl numbers for k and ϵ, C_2 and C_3 are generic constants found experimentally and for which are used usual standard values; while, at the end, we've S_k and S_ϵ that are the source terms for the turbulent kinetic energy k and for the dissipation rate ϵ, respectively.

The dynamic turbulent kinematic viscosity (or *eddies viscosity*) μ_t is defined as:

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \tag{16}$$

in which we've: the turbulent kinetic energy k, the dissipation rate ϵ and the constant C_μ related to the effects of the turbulence on the dynamic viscosity of the flow.

About the two source terms, $\dot{S}_k$ for the turbulent kinetic energy and $\dot{S}_\epsilon$, that underline the contribution of the liquid particles, we can define that, as reported also on the work by Timur Bikmukhametov, Reidar Kristoffersen and Ole Jørgen Nydal [10] and also on the Siemens methodology manual [9], as follow:

$$S_k = \sum_{l=1}^{n} (m_{lg}k - m_{gl}k); \qquad S_\epsilon = \sum_{l=1}^{n} (m_{lg}\epsilon - m_{gl}\epsilon); \tag{17}$$

with k and ϵ that are the turbulent kinetic energy and the dissipation rate of the continuous phase, so the gaseous flow.

The values used for the different constant of the previous equation, to model the turbulence, are shown on the next page.

C_μ	C_1	C_2	C_3	σ_k	σ_ϵ
0,09	1,55	1,92	0	1	1,3

Table 1: Constants used by the $k-\epsilon$ model to describe the turbulent behaviour of the gaseous phase.

3.3.5 Thermophysical models

To study the evolution of the thermophysical properties of our gaseous phase, we've chosen some thermophysical models to describe how energy, heat and physical properties evolve.

As equation of state we use the **perfect gas equation** to describe the density of the gaseous phase as function of pressure p, universal gas constant R and temperature T:

$$\rho = \frac{p}{RT} \tag{18}$$

The transport model adopted is the **the Sutherland model** that allows us to compute the dynamic viscosity μ, as function of temperature T and Sutherland temperature T_s ($170,67$ for dodecane, $182,41$ for ethanol) , with the use of a coefficent A_s ($1,67 \cdot 10^{-6}$ for dodecane, $1,82 \cdot 10^{-6}$ for ethanol):

$$\mu = \frac{A_S\sqrt{T}}{1 + T_S/T} \tag{19}$$

While the equation of state takes care of the pressure, the temperature and the universal gas constant, chosen for the gas mixture of the environment, so as an automatic process made by OpenFOAM, without specifying any constants (because the universal gas constant R is that setted for each gas mixture at a certain temperature); in the case of the Sutherland model, we've to define the values of the two constants A_S and T_S, in order to obtain the dynamic viscosity and the thermal conductivity.

The thermodynamic model computes the specific heat capacity c_p from which other properties are obtained. We use the **Janaf model** in which there is a set of *Janaf coefficients*, which can be computed from specific thermodynamics' tables.

$$c_p = R((((a_4T + a_3)T + a_2)T + a_1)T + a_0) \tag{20}$$

with R that is *the universal gas constant.*

It's important to observe that there are also two coefficents a_5 and a_6 that are used to evaluate the enthalpy h and the entropy s, respectively.

3.4 Lagrangian Phase

After seeing how the surrounding gas phase is studied, with the Eulerian approach, it's time to explain the Lagrangian approach, that reveals to be the best way to study liquid particles injected into a duct crossed by a gaseous flow.

In computational fluid-dynamics it's not possible to choose the best approach between the Eulerian method and the Lagrangian one, because each of them reveals to be the best one according to the specific situation that you'd like to study. The Lagrangian is the best one to study injected liquid particles, because it uses the particle itself as reference system: so the quantity that it allows to study, have to be considered as function of time. Each particle is as followed by the reference system, and we study how its characteristics change over time, without considering its spatial position.

As we've said at the beginning of the chapter, the three families of governing equations will always be the same ones shown previously.

3.4.1 Mass Equation

The mass equation, in this case considered through the Lagrangian method, is now applied to each spherical rigid particle (this is our approach to study the particles, that are so simplified and considered as spheres without a flow inside them). Some useful references about the equation below and how to get it are the book of S. Turns [11], the work written by Alireza Ghasemi [1] and the work of M Burger & Co [12], which explain in detail every minimum consideration that must be taken into account:

$$\frac{dm_d}{dt} = \frac{m_d}{\tau_i} \tag{21}$$

We've that m_d represents the mass of the particle/droplet and τ_i is the *relaxation time*, that, according to the model adopted, can be considered as the *evaporation* relaxation time or the *boiling* relaxation time.

The original definition of the relaxation time is taken from the book "Multiphase Flows with Droplets and Particles" [13] due to its very detailed explanation of all the physical quantities involved to compute the relaxation time.

We can, however, divide by the density and re-write the equation as function of the diameter of the droplet/particle, as follows:

$$\frac{dD}{dt} = -\frac{D}{3\tau_i} \tag{22}$$

where:

- $\tau_i = \tau_e$ under standard evaporation;
- $\tau_i = \tau_b$ under boiling conditions.

And it's possible to compute τ_e, so the relaxation time under the *standard evaporation conditions*, as follows:

$$\tau_e = \frac{m_d}{\pi D_p D_{diff} S_h \rho_v ln(1+B)} \tag{23}$$

But, in our model, we've decided to consider the boling conditions as reference model, and we'll show the final equation some rows below.

To clarify some of the numbers that we can see for computing the relaxation time, we've that Sh is the *Sherwood number*, that can be computed from the Reynolds number Re, and as function of the diameter of the droplet D_p, according to **the Ranz-Marshall (or Frössling)** correlation:

$$S_h = 2 + 0.6Re^{1/2}Sc^{1/3} \tag{24}$$

with:

$$Re_d = \frac{\rho_l \ |u_d - u| D_p}{\mu_l} \tag{25}$$

and

$$Sc = \frac{\nu}{D_{diff}} \tag{26}$$

where ν is the *kinematic viscosity* of the gas, while μ_l is the *kinematic viscosity* of the liquid phase.

The other numbers, as the Schmidt number Sc, the mass thermal diffusion coefficent D_{diff} and the coefficent B are computed as follows:

$$B = \frac{X_{v,s} - X_{v,\infty}}{1 - X_{v,s}} \tag{27}$$

$X_{v,s}$ is the mass fraction of the fuel vapor at droplet surface and $X_{v,\infty}$ is mass fraction of the fuel vapor far away from its surface.

While, under boiling conditions, the following relaxation time (from [13]) is used:

$$\tau_b = \frac{D^2 \rho_l c_{p,v}}{2k \cdot Nu \cdot ln(\frac{C_{p,v}}{h_v}(T - T_d)) + 1} \tag{28}$$

with k that is the *thermal conductivity* of the gas , Nu is the *Nusselt number*, $C_{p,v}$ is the *isobaric heat capacity* of the gas, and h_v the *mean enthalpy* of it. T_d is the temperature of the droplet, while T is the *surrounding temperature.*

Nu can be computed as follows:

$$Nu = 2 + 0.6 Re^{1/2} Pr^{1/3} \tag{29}$$

and Pr is the Prandtl number, defined as:

$$Pr = \mu \frac{c_p}{k} \tag{30}$$

So the evaporation of the droplet is estimated through the D^2 **law**, for which is possible to find a very useful and deep explanation on the book of S. Turns [11].

$$\frac{dD^2}{dt} = C_{evap} \tag{31}$$

in which the *evaporation constant* C_{evap} (from [13]) is equal to:

$$C_{evap} = -4\ S_h \frac{\rho_v}{\rho_l} D_{diff}\ ln(1+B) \tag{32}$$

3.4.2 Momentum Equation

After considering the conservation of mass of each droplet/particle, we can consider the momentum equation, that takes care of the single particle speed u_d, the density ρ, tha mass of the particle m_d, the relative velocity $|u_d - u|$ (that is the difference between the particle speed u_d and the gaseous flow velocity u). It is also important to consider the drag coefficient, whose value has to be chosen according to the situation that we consider, as shown in the sub-chapter 3.5.1, that explains in detail the drag force, among the different particle forces.

The equation that describes the momentum equation is the one below:

$$m_d \frac{du_d}{dt} = -\frac{\pi D^2}{8} \rho C_d |u_d - u|(u_d - u) + F_G + F_P \tag{33}$$

in which we've reported the *drag force*, that represents the 80% of the total force that acts on each particle. With the drag force, it's important to consider also the *gravity /buoyancy force* F_G and the *pressure gradient force* F_P, that are explained in detail some sub-chapters below (sub-chapter 3.5).

3.4.3 Energy Equation

About the energy equation, it's necessary to consider the heat exchange between the gas phase and the liquid particles: the particle receives heat from the surrounding gaseous phase and thus increases its temperature, overcoming the latent heat of evaporation, starting the process of phase change.

Some good references to these assumptions can be found in the master thesis of A Ghasemi [1], in the Phd thesis of Niklas Nordin [14], and in the work written by F P Kärrholm [15].

$$m_d \frac{dh_d}{dt} = \dot{m}_d h_v(T_d) + \pi D \cdot k_v \cdot Nu \cdot (T - T_d) f \tag{34}$$

with m_d that is the mass of the droplet, h_d is the enthalpy of the droplet h_v is the enthalpy of the gaseous phase, T_d is the temperature of the droplet, T the temperature of the gaseous phase, Nu is the *Nusselt number* and k_v is the thermal conductivity of the gaseous phase.

At the end, f is a **factor related to the heat exchange due to mass transfer** $\dot{m}$, that can be expressed as follows:

$$f = \frac{z}{e^z - 1} \tag{35}$$

with the coefficent z that summarize the other physical properties of the droplet:

$$z = -\frac{c_{p,v}\dot{m}_d}{\pi D \cdot k_v \cdot Nu} \tag{36}$$

where $c_{p,v}$ is the *isobaric specific heat capacity* of the gaseous phase.

3.5 Particle Forces

This section is dedicated to the forces acting on the liquid particles due to the interaction with the gas phase present in the duct.

In previous sections (sub-section 3.4.2), we've shown the *momentum equation*, in the lagrangian way, used to study the liquid phase. In the momentum equation we've already written the *drag force*, because it represents the most important force acting on particles; but we've also to see other forces which influence, in a more or less marked manner, the behavior of liquid particles, also depending on the type of motion present for the gaseous phase (laminar or turbulent).

Among the various forces, the *gravity-buoyancy* force and the *local pressure gradient* force deserve special mention, and we'll show them in the next rows.

3.5.1 Drag force

Drag represents the most important force (approx. 80% of the total force) because of its effects on the movement of particles, and it's function of the drag coefficent C_D:

$$F_D = C_D \frac{\pi D_p^2}{8} \rho_f (u_f - u_p)|u_f - u_p| \tag{37}$$

With the **Drag correlations**, in the case of *spherical particles*, to obtain the drag coefficent C_D, that is:

- Schiller-Naumann (1935)

$$C_D = \begin{cases} \frac{24}{Re_p}(1 + 0.15Re_p^0.687) & if \quad Re_p \leq 1000 \\ 0.44 & if \quad Re_p > 1000 \end{cases}$$

- Putnam (1961)

$$C_D = \begin{cases} \frac{24}{Re_p}(1 + \frac{1}{6}Re_p^{2/3}) & if \quad Re_p \leq 1000 \\ 0.424 & if \quad Re_p > 1000 \end{cases}$$

3.5.2 Gravity/Buoyancy and Pressure Gradient Force

- Gravitation and Buoyancy forces are considered as a single force, being one opposite to the other::

$$F_G = m_p g \left(1 - \frac{\rho_f}{\rho_p}\right) \tag{38}$$

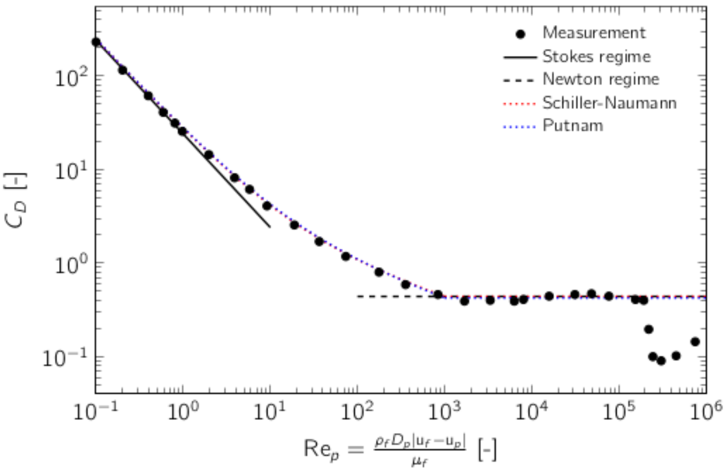


Figure 1: Drag coefficient as a function of particle Reynolds number, comparison of experimental data with correlations of Schiller-Naumann (1935) and Putnam (1961) (image from [16])

- The force due to a local pressure gradient, because of spherical particles,can calculated as:

$$F_P = -\frac{\pi D_p^3}{6} \nabla p \tag{39}$$

3.5.3 Other Forces

- **Added mass force:** the acceleration/deceleration of the particle due to the acceleration/deceleration of the fluid that surrounds the particle;
- **Thermophoretic force:** the force created by negative temperature gradients that moves fine particles;
- **Slip-rotation lift force:** lift force that acts on particles because of their free rotation in a flow (this force is also called "Magnus force");
- **Slip-shear lift force:** a tranverse lift force due to the non uniform relative velocity over the particle and the resulting nonuniform pressure distribution, that acts on particles moving in a shear layer.

3.6 Particle Response Time

Particle response time is important to understand how much the particles follow the velocity changes in the flow.

It's possible to get it starting from the equation of motion, considering only the drag force:

$$\frac{du_p}{dt} = \frac{18\mu_f}{\rho_p D_p^2}\frac{C_D Re_p}{24}(u_f - u_p) \quad \rightarrow \quad \frac{du_p}{dt} = \frac{1}{\tau_p}(u_f - u_p) \tag{40}$$

So, finally, the **particle response time**, and the **Stokes number**, can be obtained by:

$$\tau_p = \frac{\rho_p D_p^3}{18\mu_f f_D}, \qquad St = \tau_p/\tau_f \tag{41}$$

with the Stokes number St that is the ratio of the particle response time and a characteristic time scale of the flow. In the equation of the *particle response time* we've the contribution of the particle density ρ_d, the particle diameter D_p, the *dynamic viscosity* of the gaseous phase, and the constant f_D found experimentally.

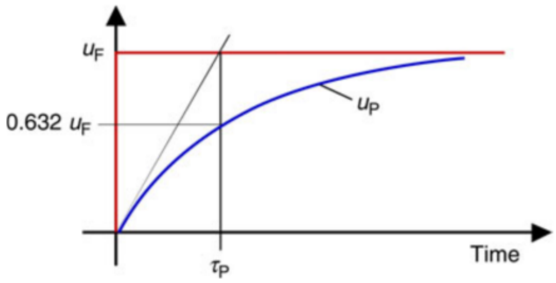


Figure 2: Graphical illustration of the particle response time (Sommerfeld, 2011)

3.7 Phase-Coupling Mechanism

Phase-coupling mechanisms explains how the continuous phase and the dispersed one behave: the different types of coupling explain if one phase influences the other or not, or if they change each other.

- **One-way coupling:** fluid → particles;
- **Two-way coupling:** fluid $\rightleftharpoons$ particles;
- **Four-way coupling:** fluid $\rightleftharpoons$ particles + particle collisions;

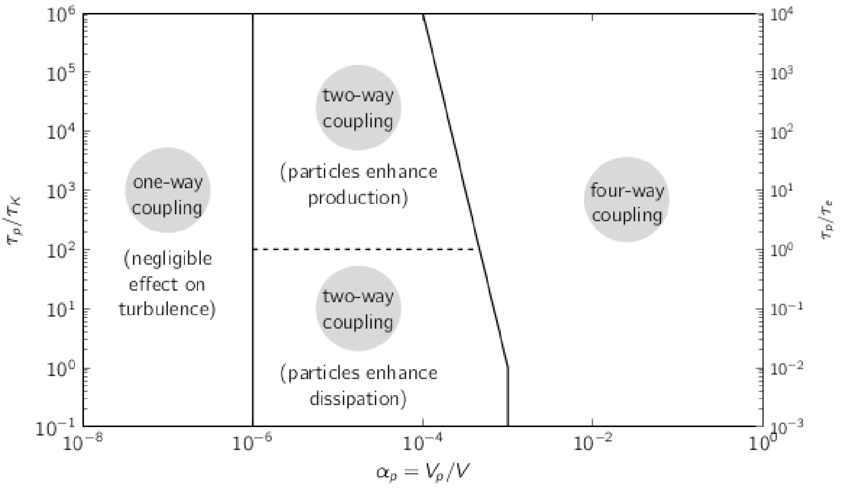


Figure 3: Classification of phase-coupling mechanisms according to Elghobashi (1994), that takes care of the ratio between the characteristic times of the two phases and of their volume ratio in a cell

3.8 Particle-Particle Interaction

To study the interaction between two or more particles, sprayFOAM uses the **soft-sphere-model (DSEM)**, that considers collisions using a spring, friction slider and dash-pot.

- <u>Normal force:</u> is obtained from the *Hertzian contact theory:*

$$F_{n,ij} = (-k_n \delta_n^{2/3} - \eta_{n,j} G \cdot n)n \tag{42}$$

where F_n stands for the *normal force*, k_n is the characteristic coefficent of the ideal spring between the two particles, δ_n the *spring displacement* along the normal direction, η the velocity of the displacement and G the ideal dashpot coefficent;

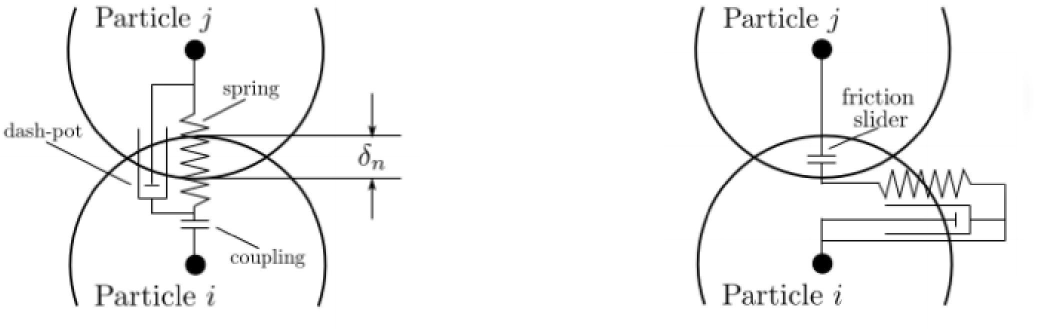
- Tangential force is computed through the following equations:

$$F_{t,ij} = -k_n \delta_t - \eta_{t,j} G_{ct} \tag{43}$$

or

$$F_{t,ij} = -\mu|F_{n,ij}| \qquad if \qquad |F_{t,ii}|_j > \mu|F_{n,ij}| \tag{44}$$

where F_t stands for the tangential force, G_{ct} is the ideal dashpot coefficent for the tangential direction and μ the friction coefficent on the particles surface.



Normal force **Tangential force**

Figure 4: The study of the forces between particles: on the left the normal force, while on the right the tangential force.

3.9 Sthocastic Dispersion Model

In order to study and predict, in an approximate way, the dispersion of particles due to turbulence, in a fluid phase, it's necessary to use the stochastic tracking model. A good explanation of this model and how CFD softwares use it are given by the Ansys User manual [8] and by the article published by Jundi Xu, M. Mohan, E. Ghahramani and H. Nilsson [17].

The dispersion model includes the effect of instantaneous turbulent velocity fluctuations on the particle trajectories through the use of stochastic methods. It considers also the particle cloud, and thus it tracks the statistical evolution of a cloud of particles about a mean trajectory.

3.9.1 Turbulence effect on particle dispersions

In the sub-chapter 3.5 we've already shown the forces that act on each particle and thus we can conclude that the irregularity of those forces, or the different directions of them, contribute, solving the force balance, to the particle motion, and so to the dispersed phase.

In turbulent flows RANS method, coupled to Lagrangian Particle Tracking (LPT) seems to be the most flexible and available way to solve and to simulate all cases.

It's possible to recall the decomposition of the instantaneous variables according to the assumption of the Reynolds-averaged Navier-Stokes equations: they can be seen as the sum of a mean value and of a fluctuating one, i.e.

$$U_i = \overline{U_i} + U_i' \tag{45}$$

$$p = \overline{p} + p' \tag{46}$$

that brings us to the $k - \epsilon$ model.

3.9.2 Stochastic model

Although the RANS method is the best one to obtain results of mean particle diffusion, it is not possible to accurately assess the value of instanteneous fluid velocity $U = \overline{U_i} + U_i'$. Thus, it's necessary to consider the fluctuating velocity U' by stochastic model, that estimates the effects of the turbulent dispersion phenomenon.

Lagrangian diffusion models take care of the mean flow properties from the RANS solution, in order to get a large number of particle trajectories and compute mean diffusion statistics. The flow field is computed later and after this a large number of particles are sprayed in the continuous field, with their trajectories determined observing each particle during its motion, until it's removed from the flow stream or from the computational domain.

We've three main categories to be listed for the Lagrangian stochastic computational models. They are:

1. Discrete Random Walk (DRW) models;
2. Continous Random Walk (CRW) models;
3. Stochastic Differential Equation (SDE) methods.

SDE methos reveal to be more complex, because they employ the so *Langevin equation* to compute the istantaneous velocity, and require modelling of the triple-velocity moments. The DRW models and the CRW ones, instead, simulate turbulence by basing on the root-mean-square values of the turbulence the functional form of the velocity perturbations.

3.9.3 Discrete Random Walk

Now let us focus our attention exclusively on the model used in our simulation: the discrete random walk (DRW) model, or "Eddy Interaction Model" (EIM). This model simulates how a particle interacts through a succession of discrete fluid phase turbulent eddies. It's a popular model to describe how a particle disperses into a turbulent flow: the particles interact with the instantaneous velocity field; and this interactions is described thanks to a stochastic random walk treatement.

This model assumes **isotropic eddies**: due to this the fluctuations of the eddies are the same in all direction because they don't have a preferred direction, thus:

$$U_1'^2 \approx U_2'^2 \approx U_3'^3 \tag{47}$$

while, for the instantaneous part of the velocities, we've that:

$$U_1' \neq U_2' \neq U_3' \tag{48}$$

which cause large over-predictions of particle deposition rates.

$$\tau_e = 2T_L \tag{49}$$

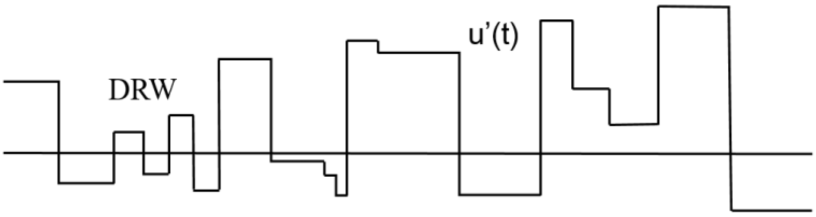


Figure 5: Velocity fluctuation time histories for the DRW

This model considers the turbulent dispersion of particles as a sum of interactions between a particle and eddies.

About the eddies: we've that their length is finite and also their existence time, according to the turbulent dispersion of particles model, that is based on the model of Gosman and Ioannides:

$$L_e = (C_\mu)^{3/4} \frac{k^{3/2}}{\epsilon} \tag{50}$$

with T_L that is the *Lagrangian integral time* that is the integral time scale for tracer particles:

$$T_L = C_L \frac{k}{\epsilon} \tag{51}$$

Using the RANS method, in particular the $k - \epsilon$ model, the mean velocity and the fluctuating one components are based on **randomly distributed Gaussian variables** whos root mean square values are equal and deduced from the turbulent kinetic energy k:

$$\sqrt{\overline{U_1'^2}} = \sqrt{\overline{U_2'^2}} = \sqrt{\overline{U_3'^2}} = \sqrt{\frac{2k}{3}} \tag{52}$$

Whereas the fluctuating velocity is computed by:

$$\begin{aligned} U_1' &= \lambda_1\sqrt{\overline{U_1'^2}} \\ U_2' &= \lambda_2\sqrt{\overline{U_2'^2}} \\ U_3' &= \lambda_3\sqrt{\overline{U_3'^2}} \end{aligned} \tag{53}$$

with λ that are *Gaussian random variables*, for which we've a nil mean and a unit standard deviation.

It's important to underline, for all the DRW model, the **isotropic turbulence assumption**, which is usually not very realistic because turbulence is a very anisotropic phenomenon, but in the case of a spary, which has no boundary layers, it is reasonable to suppose that the turbulence is isotropic.

3.10 Droplets Breakup

The interaction of a liquid droplet with the gaseous phase that surrounds it, due to the forces that take part to this dual phase-coupling mechanism, is important to be considered because it has effects on the two phases: in this case, in which the liquid particles are considered as rigid spheres, the deformation of each particle can be neglected, but we've also to consider how these forces bring the liquid particles to divide or join together, giving rise to phenomena of rupture.

In addition to the viscous forces, it's important to take care of the aerodynamic forces and the surface tension force. More detailed papers about these forces and how they influence the droplet breakup are the work by Mohit Jain, R.SuryaPrakash, Gaurav Tomar1 and R.V. Ravikrishna1 [18] and the paper *"Analytical Investigation of Droplet Breakup in Accelerated Flow"* [19].

Two dimensionless numbers that are fundamental in the study of the interaction between the two phases, are the **Weber Number** We and the **Ohnesorge number** Oh:

- Weber Number (We): it's the ratio between the aerodynamic forces exerted by the surrounding gaseous phase and the surface tension force:

$$We = \frac{\rho_v u_{slip}^2 D_p}{\sigma} \tag{54}$$

where ρ_v is the density of the gaseous phase that surrounds the liquid particle, u_{slip} is the difference in velocity between the two phases, D_p is the diameter of the particle and σ is the surface tension force exerted by the gaseous phase on the liquid particle.

- Ohnesorge number (Oh): it represents the ratio between the viscous forces and the interia-surface tension force. Thus, it underlines the viscous forces effect:

  $$Oh = \frac{\mu_l}{\sqrt{\rho_l D_p \sigma}} \tag{55}$$

  where μ_l is the dynamic viscosity of the liquid phase.

Therefore, it is possible to say that a higher Weber number represents higher external forces acting on the droplet, forcing its break. On the contrary, with a lower Weber number we've that the surface tension is greater than the aerodynamic forces, so we've a more stable situation, that doesn't bring the particle to break up.

Higher Ohnesorge numbers mean more droplet stability: the viscous forces into the droplet use the kinetic energy and so they dampen the oscillations, which could cause the drop to break.

3.10.1 Primary breakup

In the atomization process we can distinguish two types of breakup related to the liquid phase injected into the environment: the primary breakup and the secondary breakup. Primary breakup occurs at the injection, when the liquid flow leaves the nozzle. This primary breakup is influenced by the effects inside the nozzle (e.g. cavitation) and by the transition from the nozzle to the dense spray region.

It's difficult to establish a good model to approximate the effects of this liquid core breakup, but, thanks to the high injection pressure (that in our spray simulation is equal to 150 MPa) and the density of the spray, the primary breakup can be considered as done, without describing the passage from the liquid flow to the liquid particles. This assumption is justified by the study carried out by Smallwood G J, Gülder Ö L and Snelling D R [20].

3.10.2 Seconday breakup: the KHRT breakup model

As break-up model, to describe how our spray behaves after the injection, the best and most accurate choice is that of the **KHRT break-up model**, where the acronym KHRT stands for Kelvin-Helmotz instabilities (KH) combined with Rayleigh-Taylor instabilities (RT).

This break-up model is a composite process in which we calculate:

- **Kelvin-Helmotz aerodynamic instabilities** that grows on a particle surface;
- **Rayleight-Taylor instabilities** that resuts from the deceleration of the injected particles.

The two physical models act together and the one predicting the fastest onset of an instability causes the break-up. The characteristics of the two models are:

- Kelvin-Helmotz model: small droplets are shed by the parent computational parcel to form a new parcel;
- Rayleigh-Taylor model: all droplets break up completly into new droplets of different size.

In the Rayleigh-Taylor break-up model we've that the number of droplets is adjusted to keep the same mass, and to keep the original number of parcels.

For the Kevin-Helmotz breakup model, when a droplet has a diameter larger then the wavelength λ_{KH} of the wave that creates instability on its surface, it breaks into a new pair of droplets: a big droplet and a small one. The diameter of the smaller one, that represents also a "stable" droplet (means that it does not risk breaking again, unlike the larger one) is equal to:

$$D_S = 2B_0\lambda_{KH} \tag{56}$$

where B_0 is a constant assumed into the model. (typical value $B_0 = 0.61$)

The *Reitz-Diwakaer model* defines the rate of change of the bigger droplet diameter D:

$$\frac{dD}{dt} = -\frac{D - D_S}{\tau_{KH}} \tag{57}$$

with the τ_{KH} that is the *characteristic break-up timescale* and can be obtained thorugh the relation suggested by Reitz and Senecal:

$$\tau_{KH} = \frac{3.726 B_1 D/2}{\lambda_{KH}\Omega_{KH}} \tag{58}$$

where B_1 is a constant assumed into the model (typical value $B_1 = 40$, while Ω_{KH} is the maxium *growth rate* for an instability wave.

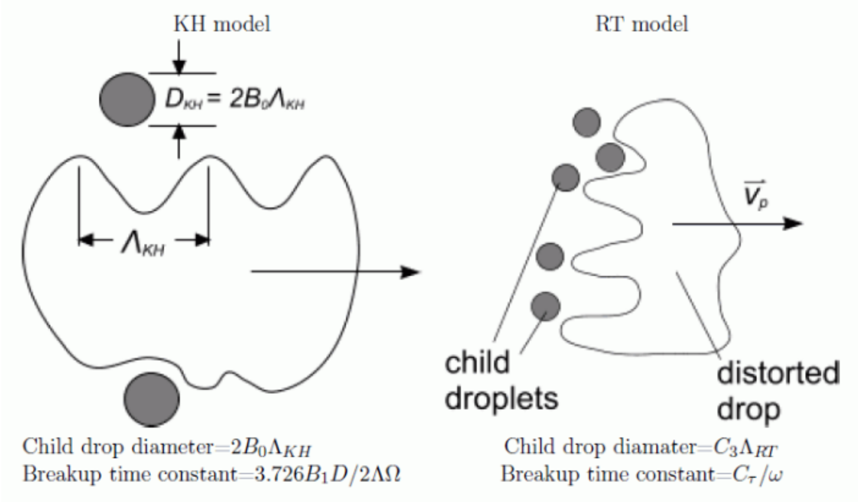


Figure 6: KHRT break-up process schematic (image from [9])

Obviously, with the droplet diameter that shrinks, the mass of the droplet decreases. If the mass of the new particle is less than 3% less than the mass of the original particle, the smaller particle is eliminated to create a new particle.

The RT break-up model occurs, instead, when the droplet diameter D is larger that the wavelength λ_{RT}, in which the constant C_3 (usually 0.1) is used to scale it. This condition can be summed up as:

$$D > C_3\lambda_{RT} \tag{59}$$

Now, satisfied this condition, it's necessary to spend a sufficient time larger then the RT break-up timescale τ_{RT}. To obtain λ_{RT} is needed to calculate the corresponding wave number $k_{RT} = 2\pi/\lambda_{RT}$, wich maximizes the growth rate, that can be obtained by:

$$\omega(k) = -k^2\Big(\frac{\mu_l+\mu_g}{\rho_l+\rho_g}\Big) + \sqrt{k\Big(\frac{\rho_l-\rho_g}{\rho_l+\rho_g}\Big)a - \frac{k^3\sigma}{\rho_l+\rho_g} + k^4\Big(\frac{\mu_l+\mu_g}{\rho_l+\rho_g}\Big)^2} \tag{60}$$

thus, the RT break-up timescale is obtained by:

$$\tau_{RT} = \frac{C_\tau}{\omega_{RT}} \tag{61}$$

with $\omega_{RT} = \omega(k_{RT})$ and C_τ that is a constant of the model (typical value $C_\tau = 1$).

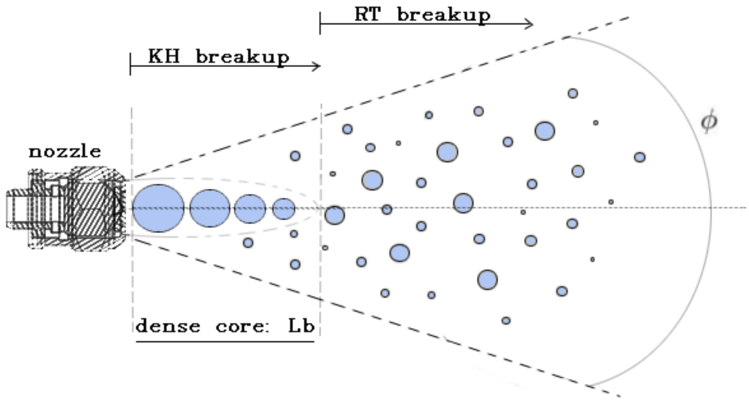


Figure 7: Breakup schematics in KH-RT model (idea from [1])

.

3.10.3 Another secondary breakup model: ReitzDiwakar model

In order to be clear in our study, it's necessary to define the other secondary breakup model that we'll analyze in the subchapter 6.6, in order to get the main differences with respect the one that we've taken as reference model (the KHRT breakup model). Good references to analyze in more detail what is written below are the work of S Hossainpour and A R Binesh [5], the "Siemens Methodology" manual [9] or the paper written by P Ghadimi, M Yousefifard and H Nowruzi [6].

The other model, that is explained as next, is the *ReitzDiwakar* breakup model: it distinguishes, as the KHRT breakup model, two breakup regimes, that are listed below:

- **Bag breakup:** it occurs for low relative speeds with respect to the gas phase;
- **Stripping breakup:** it is considered for high relative speeds with respect to the gaseous phase.

When the breakup is happening, regardless of the breakup regime that occurs, the reduction of the unstable particle radius is obtained as follows:

$$\frac{dr_p}{dt} = -\frac{(r_p - r_{stable})}{\tau_{br}} \tag{62}$$

where r_p is the radius of the unstable droplets, r_{stable} is the radius that the droplet will obtain at the end in the stable condition and τ_{br} is the characteristic time of the breakup regime that is occured.

The Bag breakup occurs when the Weber number We results to be greater then the *critical Weber number for bag* We_{crit} (typical value $We_{crit} = 6.0$):

$$We = \frac{\rho_v u_{slip}^2 D_p}{\sigma} \geq W_{crit} \tag{63}$$

and, when we're in this breakup regime, we've that the *characteristic time* $\tau_{br,bag}$ and the *radius of the stable droplet* $r_{stable,bag}$ are equal to:

$$\tau_{br,bag} = C_{bag}\sqrt{\frac{\rho_l r_p^3}{2\sigma}} \qquad r_{stable,bag} = \frac{6\sigma}{\rho_v u_{slip}^2} \tag{64}$$

where C_{bag} is the *time factor* for bag breakup (typical value $C_{bag} = \pi$), σ is the surface tension force on the particle, ρ_l is the density of the droplet (the liquid phase) and u_{slip} is the *slip velocity* between the liquid phase and the gaseous one.

The Stripping breakup occurs, instead, when the ratio between the Weber number We and the square root of the Reynolds number Re is higher then a model constant C_{s1} that represents the *Weber number factor for stripping* (tpycal value $C_{s1} = 0.5$):

$$\frac{We}{\sqrt{Re}} = \frac{\rho_v |u_d - u| \ \mu}{\rho_l \sigma} > C_{s1} = 0.5 \tag{65}$$

with μ that represents the dynamic viscosity of the gaseous phase.

In the stripping breakup regime, the characteristic time $\tau_{br,strip}$ and the radius of the stable droplet $r_{stable,strip}$ are equal to:

$$\tau_{br,strip} = C_{strip} \frac{r_p}{u_{slip}} \sqrt{\frac{\rho_l}{\rho_v}} \qquad r_{stable,strip} = \frac{\sigma^2}{2\rho_v^2 u_{slip}^3 \nu} \tag{66}$$

where C_{strip} is the *time factor* for stripping breakup (typical value $C_{strip} = 20$ and ν is the *kinematic viscosity* of the gaseous phase.

3.10.4 Why didn't we use other secondary breakup models?

In addition to the two models explained above, others are available to approximate and study the secondary breakup that affects the liquid particles injected into the gas phase: OpenFoam guarantees us the possibility of using other models, which for various reasons we have decided not to explain in detail and not to use in the comparison of the subsection 6.6:

- **Pilch and Erdman breakup model:** it was impossible to use this very interesting model, as it presents obvious calculation errors in the source code;

- **TAB breakup model:** The criticism of this model is that it predicts very small droplets, and they form like a "finger" (a prevalent concentration in the line passing throug the nozzle oultlet), so without forming the conical shape, as setted by our model of injection. Gas flow in these types of spray is from the outside towards the centerline of the spray: the small droplets are brought to the centerline while the large droplets have a bailistic trajectory: these mechanism causes a very "fluctuating" trend, that passes from values that tend to be normal to null values in a few moments (it's connected to the injection rate).

The reasons given above are the result of research and consulting work at the CFD research forum (http://www.cfd-online.com)

3.11 Rosin-Rammler Distribution

As we've specified in 3.10.1, the liquid core could be assumed as negligible. In this way, it's possible to direclty implement the result of primary breakup in the nozzle outlet through a *droplet size distribution.*

We've chosen, in order to describe the dispersed particles in the flow, a standard **Rosin-Rammler distribution.**, whose original explanation is reported on [7].

To describe the cumulative distribution function, we've the following equation:

$$F(D) = 1 - exp\Bigg[-\left(\frac{D}{d}\right)^{n}\Bigg] \tag{67}$$

and, by differentiating it, we obtain the probability density funcion:

$$f\,(D) = -\frac{n}{D}\left(\frac{D}{d}\right)^{n} exp\Bigg[-\left(\frac{D}{d}\right)^{n}\Bigg] \tag{68}$$

The functions have two input parameters:

- The n-parameter, which is a measure of the spread in particle sizes, where a higher value reduces the spread.

- The d-parameter is the mean particle size of the distribution. It corresponds to the average droplet size of the probability density function.

About the droplet size, we've that the minimum and the maximum size values have to be given in OpenFOAM, in order to normalize the probability density function with a $K - factor$:

$$K \int_{D_{min}}^{D_{max}} f\,(D)dD = 1 \tag{69}$$

$$K = \frac{1}{exp\left[-\left(\frac{D_{min}}{d}\right)^{n}\right] - exp\left[-\left(\frac{D_{max}}{d}\right)^{n}\right]} \tag{70}$$

3.12 Particle size monitoring

In order to monitor particle size distribution there are several methods, and it is important to explain those used in this work, to understand the study made for the breakup models comparison (in subsection 6.6).

To follow the particle that are parts of the spray, OpenFOAM takes care of three different quantities:

- $\boldsymbol{D_{10}}$ represents the diameter at which 10% of the total mass is composed by particles with a diameter smaller then D_{10} itself;
- $\boldsymbol{D_{32}}$ means the *Sauter Mean Diameter*
- $\boldsymbol{D_{max}}$ is just the maximum droplet diameter at a certain time step.

3.12.1 Sauter Mean Diameter

It's an average of particle size that in fluid dynamics is used especially in the study of a liquid spray. It's computed through two quantites: the surface diameter d_s and the volume diameter d_v:

- surface diameter d_s is defined as follows:

$$d_s = \sqrt{\frac{A_p}{\pi}} \tag{71}$$

  where A_p is the surface area of the particle;

- volume diamer d_v is equal to:

$$d_v = \left(\frac{6V_p}{\pi}\right)^{1/3} \tag{72}$$

  where V_p is the volume of the particle

Thus, the Sauter mean diameter (SMD), or the d_32 diameter can be expressed as follows:

$$D_{32} = \frac{d_v^3}{d_s^2} \tag{73}$$

Once the formula is defined, we show the final analytical formula used in the case of a spray:

$$D_{32} = \left(\frac{\sum_{i=1}^{n_p} d_{v,i}^3}{\sum_{i=1}^{n_p} d_{s,i}^2}\right) \tag{74}$$

where n_p is the total number of particles.

4 Numerical methods

4.1 OpenFOAM and SprayFoam

This is the last chapter of the theoretical part of the work, in which is good to explain, not in a too much detailed way, how *OpenFOAM*, in particular the solver *sprayFoam*, works numerically with the physical models that are reported in the two previous chapters. All the results obtained and all the simulations done come from the latest version of OpenFOAM, the version 6.

The task of the following reasearch work is to analyze the behaviour of a liquid spray injected into a square duct, through which there is a flow of air with a certain velocity.

OpenFOAM has a large number of solvers, that might be chosed according to the physical situation that you would like to describe and to simulate.

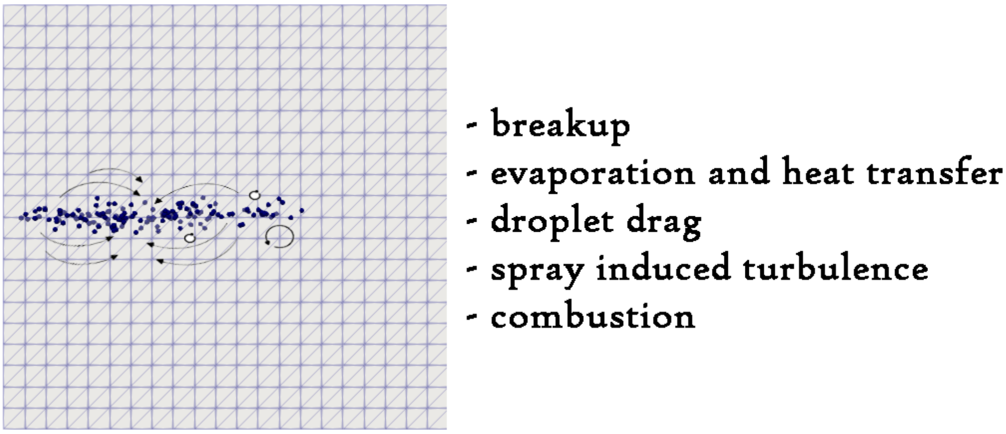


Figure 8: The several phenomena that occur with a spray

On the picture above are represented the most important phenomena that occur within a spray, due to the interaction with a gaseous phase surrounding it.

All these phenomena are described in the theoretical part of this work, into the previous chapter, that makes a good introduction to understand in a good way the phenomena involved and that the CFD tool has to represent, with all the parameters involved.

SprayFoam represents a good choice because it's a transient solver for compressible, turbulent flows with a spray particle cloud; that can be customized with an advanced knowledge of fluid dynamics and of the C++ programming language, that represents the language by which the source code of OpenFOAM is written.

4.2 Relationship between the lagrangian study and the eulerian one

It's important to focus the attention on how the spray particle are treated to be studied through the LPT method (in sub-chapter 4.3) and how the interaction between the two phases is intended (in subchapters 4.3.1 and 4.3.2). We've also to define what is the PIMPLE algorithm (in sub-chapter 4.3.3) that is used in several CFD softwaretools, and, in this case, allows us to couple the velocity of the flow and of the spray which is present for both of them in the *Navier-Stokes equations* (in sub-chapter 3.3 for the gaseous flow and in sub-chapter 3.4 for the liquid particles).

And to finish this numerical description of the work, it's also important to give a definition of the *Courant number*, a fundamental parameter (explained in sub-chapter 4.4) in the field of computational fluid-dynamics, and of the *bwUnicluster* platform (in sub-chapter 4.5), that allows the use of powerful resources to do some computational calculations.

4.3 Parcel and Spray Concept

In the development of the spray injection and its evolution through the duct, we've to consider how sprayFoam treats the particles to apply the Lagrangian-Paricle-Tracking (LPT) method.

A certain number of parcels composes the spray, that is injected into the domain. Each parcel is then composed by a certain number of particles. Introducing a fixed number of parcels eliminates the need to resolve the equations for every droplet, reducing the computational load. In choosing the number of parcels, it's important to consider the right balance between accuracy and computational burden. If a low amount of parcels is used, droplets with different characteristics will be included in the same group and considered to be similar, leading to erroneous results.

The definition given by OpenFOAM of these two classes (spray and parcel) is:

- **Spray:** it represents many parcels with varying parcel properties;
- **Parcel:** it represents many particles with identical particle properties.

The gas medium is treated in an Eulerian fashion and the spray evolves into the domain based on the mass, momentum and energy exchange with the continuous gas phase. Thus, the two phases interact with each other in order to determine the evolution of the spray and the mixture formation. Thus, this subdivision is made by OpenFOAM in order to set the *Eulerian-Lagrangian Coupling.*

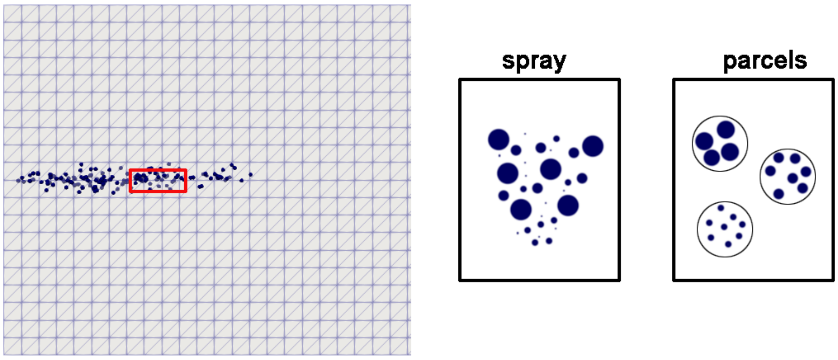


Figure 9: Representation of the two classes in which the particles are grouped.

4.3.1 Gas-Liquid Coupling

This type of coupling describes the interpolation process so that the gas properties known at the Eulerian nodes are estimated at the location of the parcels. A more detailed explanation of what is reported here can be found in A. Ghasemi [1], in R. Kasper [21] and in F.P. Kärrholm [15].

Firstly, for the velocity field, known at the cell centers, is evaluated at the closest vertices or faces. Thus, we've a tetrahedron built around the parcel.

Looking at the image below, it's possible to see that α_i represents the distance between the parcel and one of the tetraheron faces, while $\alpha_i+\beta_i$ is the distance between the same face and its opposite vertex.

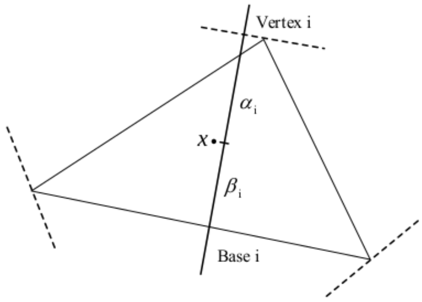


Figure 10: Representation of the parcel position interpolation into the Eulerian fashion.(image from [1])

Interpolating it's therefore possible to obtain the velocity of the parcel at position x:

$$u(x) = \sum_{i=1}^{4} \frac{\beta_i}{\alpha_i + \beta_i} u_i \tag{75}$$

Using this interpolation method, we've that each parcel experiences a continous velocity field.

4.3.2 Liquid-Gas Coupling

The liquid-gas coupling represents the effect of spray source terms on the Eulerian governing equations.

We've a certain temporal "shift" between the Eulerian solution and the parcels: the first one is frozen at time instant n, while the parcels are advanced one by one to the next time step $n+1$.

The face-to-face algorithm is used to track each parcel along its path, while all the source terms for mass, momentum and energy equations are evaluated at time n. In this way we can identify all the cells that are crossed by each parcel during one time step and to split the Lagrangian source terms of the Eulerian euqations.

For example, with a velocity of the parcel lower than the interpolated Eulerian velocity, there will be a reduction of the Eulerian velocity and momentum gained by it. Also the energy equation accounts for heat tranfer and evaporation of the droplets.

The injection of the parcels starts from a hole/disk that represents the exit side of a conical nozzle and the points are chosen and organized according to the Rosin-Rammler distribution, as we've specified before. The frequency of addition of new parcels is directly related to the fuel total mass and to the number of parcels per second.

4.3.3 Pressure and Velocity Coupling

Looking at the Navier-Stokes equations is possible to see that an explicit equation for the pressure is not available. In order to remedy this lack, a coupling scheme to relate the pressure values to the velocity field is used. The algorithm used by sprayFOAM is a PIMPLE algortihm, which is a coupling of SIMPLE and PISO algorithms.

A good explanation of the main solvers used by CFD tools can be found easily on any site that deals with fluid dynamics studies. In this case, we've reported the definition of these solvers given us by the OpenFOAM User Guide [2]:

The PISO algorithm stands for Pressure-Implicit-Split-Operator, while the SIMPLE algortihm stands for Semi-Implicit-Method of Pressure-Linked Equations. These ones, with the PIMPLE algorithm, that is a combination of the two ones, couple together equations for mass and momentum conservation through iterative procedures. In cases of transient problems PISO and PIMPLE algorithms are used, while for steady-state the SIMPLE algorithm is used.

In case of transient problems, PISO and PIMPLE solve a pressure equation, considering mass conservation, with an explicit correction to velocity to satisfy momentum conservation. Optionally, they can solve the momentum equation, each step, taking care of the so-called "momentum predictor."

The looping is controlled by **input parameters**, that are listed below, reporting their definition given by the OpenFOAM Userguide:

- ***nCorrectors:*** (for PISO and PIMPLE algorithms) used to set how many times the pressure equation and the momentum corrector are solved by the algorithms in each time step (typically set to 2 or 3). *(in our case equal to 2)*
- ***nNonOrthogonalCorrectors:*** (for all algorithms), defines solutions of the pressure equation for repeated times, it's used to update the explicit non-orthogonal correction of the Laplacian term $\nabla \cdot ((1/A)\nabla p)$; (typically 0, particularly for steady-state, or 1). *(in our case equal to 0)*
- ***nOuterCorrectors:*** (for PIMPLE), it's used to set how many times the system is solved within a time step, enabling a loop over the entire system; it replaces the PISO algorithm when is set to 1, but typically it's ≥ 1. *(in our case disabled)*
- ***momentumPredictor:*** it's used to solve the momentum predictor. It controls this solution acting as a switch, that is set to off for some flows, like those in which there is a low Reynolds number or in case of those that are multiphase. *(in our case enabled)*

4.4 The Courant/CFL Number

The Courant Number is an important index to study the stability of a simulation and to regulate the transmission of information during the fluid dynamics simulation

The CFL (Courant-Fredrichs-Lewy) condition tells us that the Courant Number has to be lower than a certain value (that depends on the type of simulation that we want to perform), otherwise the simulation produces incorrect results.

The principle related to that condition is that, for example, if an information (ideally a wave) is moving across a discrete spatail grid and we want to compute its dimension (or another property) at discrete time steps of equal duration, then this duration must be less than the time for the particle to move between two adjacent points.

Thus, when the grid point separation is changed, the upper limit for the time step also changes.

For the n-dimensional case, the Courant number has the following form:

$$C = \Delta t \cdot (\sum_{i=1}^{n} \frac{u_{x_i}}{\Delta x_i}) \leq C_{max} \tag{76}$$

about the formula, we've:

- u_{x_i} : the magnitude of the velocity in the x_i direction;
- Δt : the time step;
- Δx_i : the length of the control volume in the x_i direction.

The value of C_{max} changes with the method used to solve the discretised equation, especially whether the method is explicit or implicit.

In our simulation, we've defined a value C_{max} = 0,3.

4.5 Simulations on the bwUnicluster

The last sub-chapter of the numerical methods part focuses the attention on the instrument that was used to perform some heavy calculus, necessary in some parts of this work in which we used a large number of cells for our grid, requiring a too high computational cost for a normal computer hardware.

The word "cluster" means a group of directories connected among each other through a telematic network. Thanks to this setup, we're able to use a large number of processors, that are not usual for a normal commercial computer: using a larger number of processors allows to develop a finer mesh, thus with a large number of points and cells, to decrease the time step of the simulation or to increase the computational domain.

The cluster has a large domain with different folders that are divided into different categories and sub-categories: one of this sub-categories is the one that groups all the directories of the cluster users: each user is assigned a folder. In this folder all the work files are contained and from time to time it will be possible to send in queue a job, assigning them a certain number of processors cores and also a certain degree of importance to be able to give them a given precedence in the queue.

OpenFOAM has specific functions to make use of this large number of processors: one of them is the function *decomposePar* that allows to divide the computational work among the various processors available: it's possible to assign a certain number of processors cores to each spatial direction (x,y and z). In our case, we've used an large number of processors cores, 16, dividing them by assigning 8 cores in the x-direction (that is the direction of movement and also the longest one of the computational domain) and the same number of processors for the y and z-direction (4 cores), that have the same importance in our computational work and also the same geometrical length.

Once you have chosen the configuration of the processors, OpenFOAM divides the mesh in a number of folders equal to the total number of cores used (for our setup) and creates them in a separate way, to do the computation for each of them. So when the work is done, it is possible to reconstruct all thorugh the *reconstructPar* command, that joins together all these folders in order to obtain the final one, that seems like the normal folder that it's possible to get without splitting the simulation work.

5 CFD Analysis of the duct with one liquid droplet

In order to study step by step the interaction of the gas phase with the liquid spray, we've reduced the liquid spray to just one single particle (droplet) injected into the duct, in order to see how the evaporation time of the single liquid particle changes according to the variation of a single parameter of the gas phase, or of the liquid spray itself, changing them step by step.

5.1 The liquid

In order to validate our simulation model, we decided to adopt, as liquid spray to be injected into the duct, the ethanol ($C2H5OH$). This choice is motivated by the availability of experimental data relating to the evaporation of a single liquid ethanol particle injected into a gaseous phase of air (made by Likun Ma and Dirk Roekaerts [3]), also by the possibility to use a existing 2-D Matlab model to study analytically the physical case of the single droplet (we've modified the model of Andreas Höfler [4]) and due to this it represents the fastest and the simplest way to validate our model.

5.2 sprayCloudProperties file

First of all, we've set ethanol (C2H5OH) as liquid phase, in order to use its chemical and thermophysical properties. We've used a droplet with a total mass of 8.86 $\times 10^{-11}$ kg to reflect the experimental data [3] that we'll use validate our simulation: the experimental data were obtained on the study of an ethanol particle with a diameter of 60 $\times 10^{-6}$ m.

In order to obtain the total mass, starting from the diameter of the particle, we've used simple analytical formulas, taking into account the density of the ethanol at the temperature set each time for the liquid phase:

$d = 60 \cdot 10^{-6}\ m$
$\rho = 783.4\ kg/m^3\ (300K)$

$$M = \rho \cdot V = \rho \cdot \frac{\pi \cdot d^3}{6} = 783.4 \cdot \frac{3,14 \cdot (60 \cdot 10^{-6})^3}{6} = 8.86 \times 10^{-11} \quad kg \quad (77)$$

5.3 Our model configuration

In the image reported below it's shown one section of our duct, which has a square section of side 20 cm with a length along the x direction of 60 cm.

The gaseous flow that runs through the duct has an initial speed of 20 m/s, while the initial speed of the injected particles is 35 m/s, in order to achieve a *slip velocity* of 15 m/s between the two phases. The domain and the thermophysical conditions were set to make the gaseous phase turbulent, that we have studied through the RANS method, using the k-ϵ approach.

This configuration was set in order to achieve the same initial conditions adopted by Likun Ma and Dirk Roekaerts [3] in the case studied by them and then be able to make a comparison to validate our model.

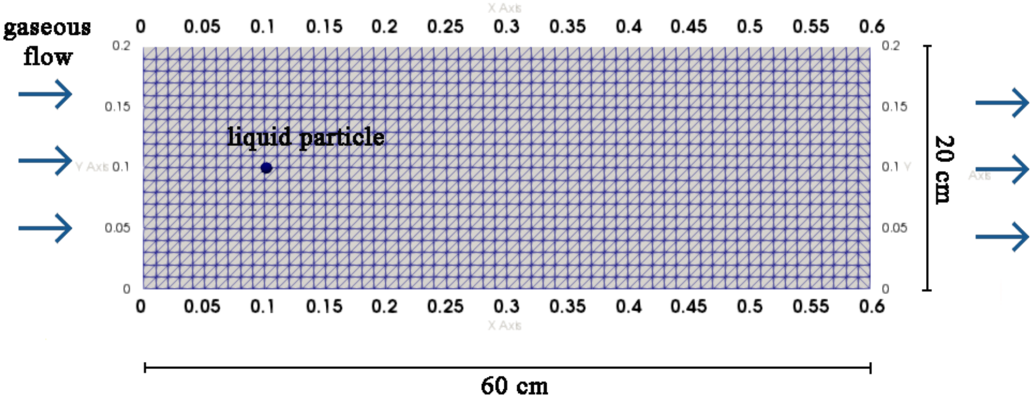


Figure 11: Graphical representation of one section of our case configuration with the geometrical dimensions and the particles injected into the duct, with the gaseous flow that runs through it.

The initial temperatuare of the droplet, again referring to the study of Likun Ma and Dirk Roekaerts, is set to be 300 K, while the surrounding temperature is changed during the study as an evaluation parameter for the time of evaporation.

5.4 Comparison with other models

To validate our model we've compared it to other models: one is that used by Likun Ma and Dirk Roekaerts [3] in their study of the single particle evaporation (that is indicated by the blue line with the title "Roekaers"), while the other one is the 2-D Matlab model of Hoefner [4], that we've modified to adapt it to our situation by setting the liquid and the initial conditions of the environment (its results are plotted through the green line with the title "Matlab").

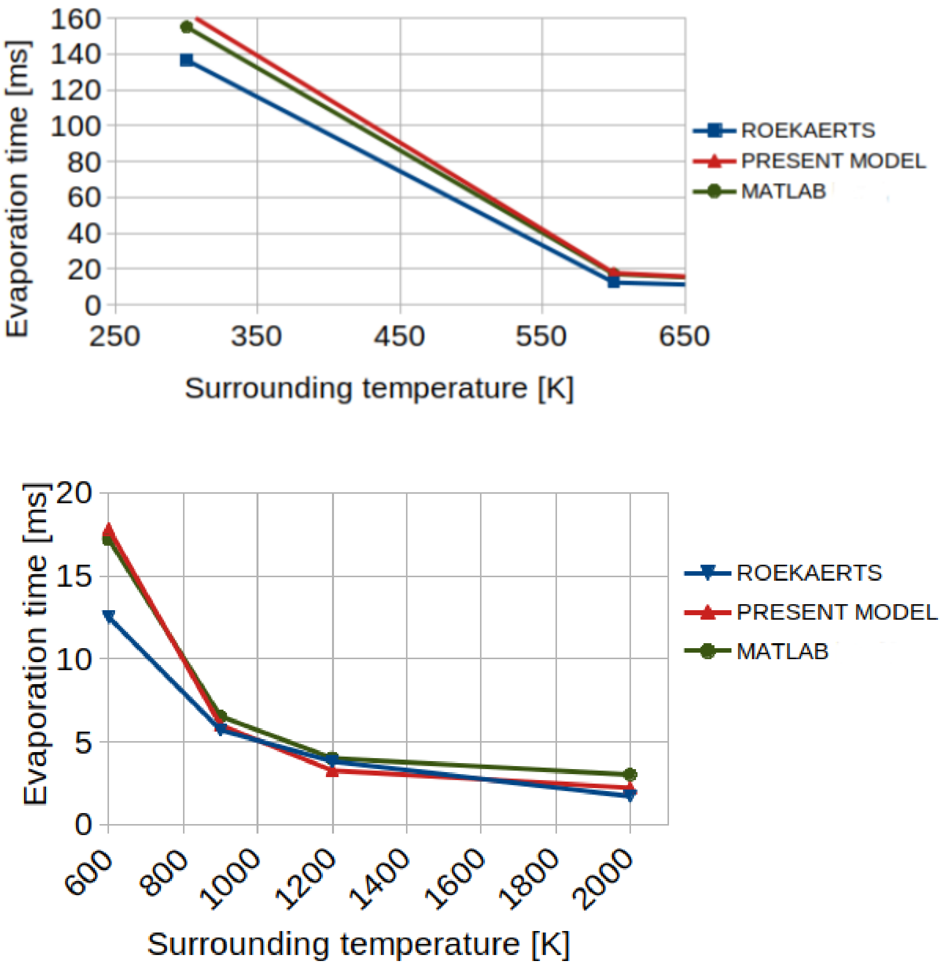


Figure 12: Evaporation time comparison between different models.

Evaporation time data comparison [ms]			
T_a [K]	Roekaerts	Present Model	Matlab
300	136,53	163	155
600	12,54	17,83	17,21
900	5,73	6	6,53
1200	3,82	3,25	4,21
2000	1,71	2,21	3,13

Table 2: Evaporation time data of the different models ($d_p = 60\mu m$)

Our comparison is based on the evaporation time of the liquid particle, changing the temperature of the surrounding environment: this represents like a "preliminary" idea of what we want to show in next chapters. The results are shown in Fig 12 in the previous page, with the Table 2 that shows the evaporation times in more detailed way.

We can easily see that the Matlab models has a bit higher values than the other two models, because, from technical reasons, they had a slightly different compostion (the presence of *urea inside the liquid droplet*).

For low temperatures our model is closer to the Matlab model, while towards the end it tends to converge towards the data developed by the Likun Ma and Dirk Roekaerts study: it's important to consider that in our case we've used a RANS turbulence model with the k-ϵ approach, while in the case of Likun Ma and Drik Roekaerts we've the use of the LES model, and this could be taken as a good point to explain why we've recorded differences in the data obtained.

Furthermore we have no information on the dimensions of the duct where the single liquid particle is injected: if it were a much wider duct than ours, this would lead to a higher turbulence intensity, explaining shorter evaporation times, given that turbulence does nothing but favor the heat transfer between the two phases (this statement is reinforced by the study of By Shashank, E Knudsen and H. Pitsch [22]).

The Matlab model, on the other hand, always has a higher value than the evaporation time due to the absence of a precise turbulence model for the gas flow, despite having the same geometric dimensions as our case.

5.5 How evaporation time changes by varying the initial conditions

Now we show as the evaporation time changes by changing a single property of the physical case time by time, in order to reach the best configuration.

5.5.1 Changing particle speed

As we said at the beginning of the chapter, to validate our model we had to simulate the same case studied by Likun Ma and Roekaerts [3], that assumes a "slip velocity", between the particle and the air flow, equal to 20 m/s in all further simulations shown, with an initial particle speed equal to 35 m/s. In order to do our comparison, we've set these values. Now, therefore, we can analyze how the evaporation time changes, by changing the speed of the particle and keeping the speed of the gaseous phase constant: in this way, by changing the particle speed we'll change the *slip velocity* between the two phases. Our results are shown in the Fig. 13 and the Tab. 3.

Evaporation time [ms]				
	$u_{particle}$ 35 m/s	$u_{particle}$ 25 m/s	$u_{particle}$ 15 m/s	$u_{particle}$ 5 m/s
$T_a[K]$	u_{slip} 15 m/s	u_{slip} 5 m/s	u_{slip} 5 m/s	u_{slip} 15 m/s
400	81,64	84,07	88,14	85,64
600	17,83	19,5	22,08	19,58
800	7,83	8,83	10,41	8,92
1000	4,75	5,42	6,42	5,25
1200	3,25	3,76	4,5	3,64

Table 3: Evaporation time data for different particle speeds

As we can see from the graph, with the values confirmed by the table, with a higher slip velocity the evaporation time has a lower value: so the situation that reveals to be more convenient to the shortest evaporatin time are the case with $u_{particle} = 35$ m/s, that has a slip velocity of 15 m/s, a higher value than the others. Thus, we continue our different simulations changing other parameters, with a velocity of the particle always assumed to be 35 m/s, and the gaseous flow velocity set to 20 m/s.

The results obtained from these analyzes are confirmed by the theoretical study carried out in the subsection about the Lagrangian phase (subsection 3.4), in which we can see how a greater flow velocity leads to a larger Reynolds number for the liquid particle (Eq. 25), which in turn increases the Sherwood number (Eq. 24), which is present at the denominator of the analytical formula for the evaporation time (Eq. 23). Consequently, an increase in the Sherwood number leads to a decrease of the liquid particle lifetime; while for the boiling condition, an increase of the Sherwood number cause the increase of the evaporation constant of the D^2 *law* (Eq. 31), so the same effect but with a different analytical equation.

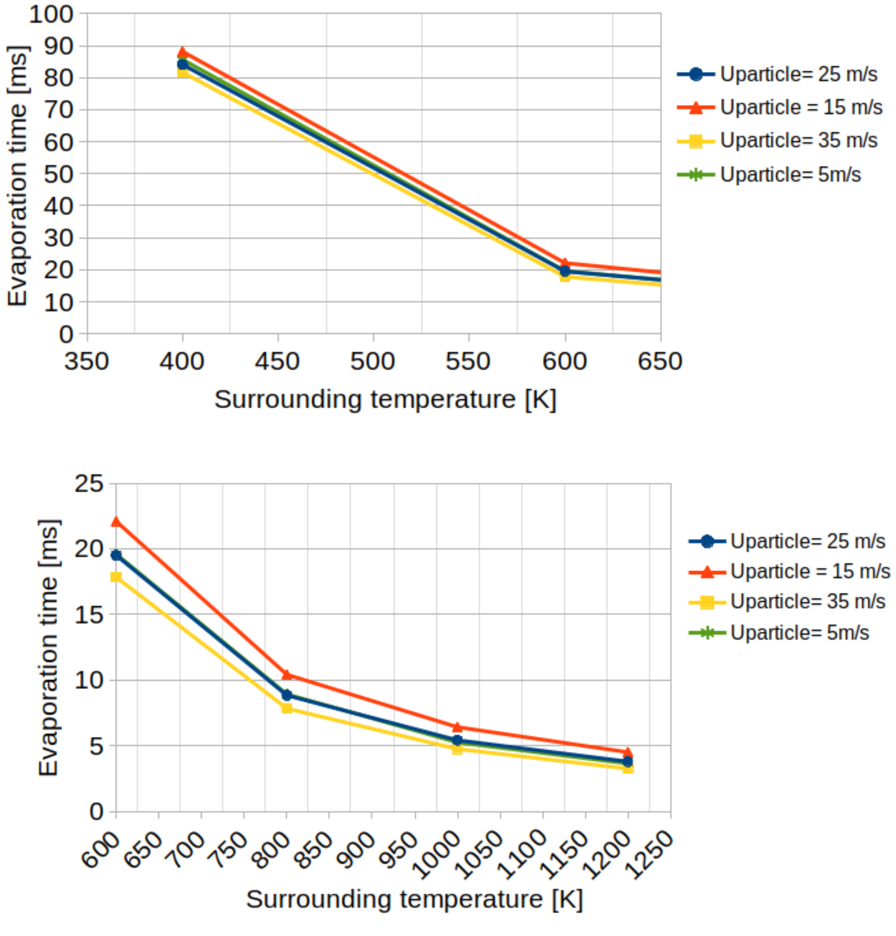


Figure 13: Evaporation time for different particle speeds and with different surrounding temperatures

As we can see, of the four different particle speeds, we have two pairs with the same slip velocity. For each couple, the evaporation time is shorter for the particle with the higher speed, without depending on the interaction with the gas: this is related to the fact the temperature and velocity has a proportional relation and so a higher velocity causes a higher temperature, so decreasing the time to reach the boiling temperature. For this reason, we've that particles with higher speed has a lower lifetime: the speed of the particle itself influences the turbulence intensity, and we know that with a higher turbulence intensity the heat propagation results to be enhanced.

5.5.2 Changing particle size

One of the parameters that we've changed to analyze its effects on the evaporation time of our liquid particle through the gas flow is the particle's size: this kind of study has been done to analyze if the results of our simulations respect what described by D^2 *law* (Eq. 31) or if there are other aspects to consider changing the particle diameter. It must in fact be considered that, conceptually, for a larger diameter there is a greater mass to be evaporated, but also a greater external surface for the particle, which therefore favors a greater heat exchange.

Evaporation time [ms]					
T_a [K]	$D_p = 20\ \mu m$	$D_p = 40\ \mu m$	$D_p = 60\ \mu m$	$D_p = 80\ \mu m$	$D_p = 100\ \mu m$
400	14,19	44,06	81,64	158	221,5
600	2	8,75	17,83	32	41,25
800	0,87	3,79	7,83	13,58	20,5
1000	0,5	2,21	4,75	7,92	12
1200	0,25	1,5	3,25	5,5	8,33

Table 4: Evaporation time data for different particle sizes

In the Figure 14 and the Table 4 we can see our results for different particle's diameters: but we can easily evaluate that the quantity of liquid mass has the greatest effect on the evaporation time that the effects from the greater external surface for the heat exchange: this is justified by the fact that the mass of the particle is at the numerator in the analytical equation for the evaporation time (Eq. 23), while the external surface is just represented by the diameter of the particle D_p, therefore a non-direct relationship, but linked to the square root of the surface, while the quantity of liquid mass has a linear relationship.

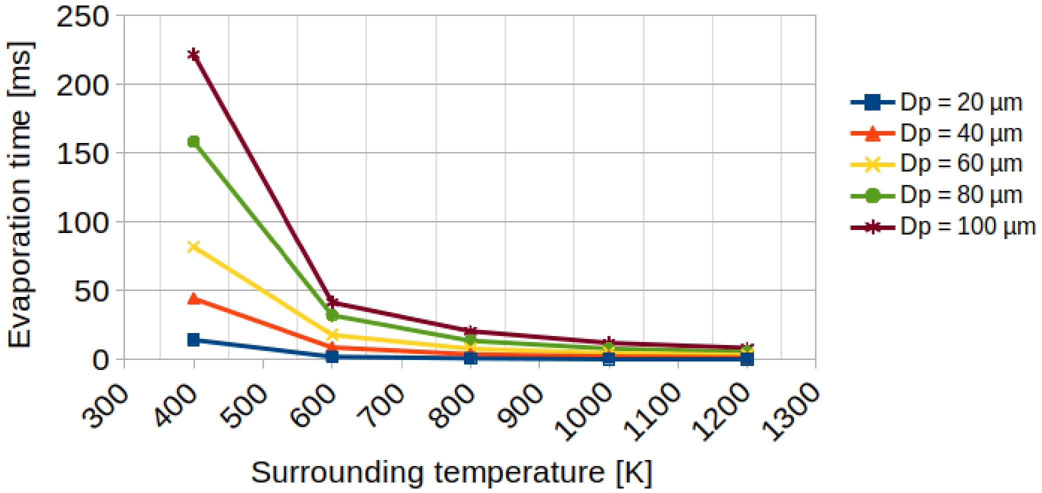


Figure 14: Evaporation time for different particle sizes and with different surrounding temperatures

5.5.3 Changing surrounding pressure

The evaluation of the surrounding pressure influence is related to the increase of the gaseous phase density, due to the proportional relation between the pressure and the density, explained by the *equation of state*, that in the case studied is that of the **perfect gas** (Eq. 18). It's important also to consider that in the formula to have the evaporation time of the droplet there is the isobaric heat capacity of the gas into the logarithmic operator (Eq. 28): this another equation that cannot be neglected. The goal of this study is to investigate if this increase of pressure has a proportional effect, without considering the total value of the pressure or not.

Let us explain ourselves better: does a pressure increase in the same way reduce the evaporation time of the particle at each temperature, or in the case of very high pressures the evaporation time is reduced in a tangible manner?

T_a [K]	0,25 bar	0.5 bar	1 bar	1,25 bar	2 bar
600	20,25	19	17,83	17,42	16,75
800	9,42	8,67	7,83	7,67	7,17
1000	5,75	5,25	4,75	4,58	4,29
1200	4,08	3,64	3,25	3,12	2,87

Table 5: Evaporation time data for different surrounding pressures

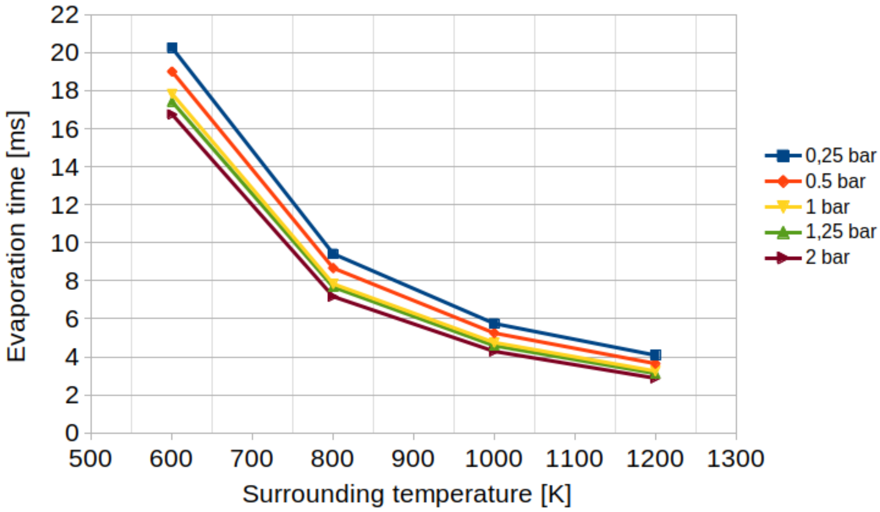


Figure 15: Evaporation time for different surrounding pressures and with different surrounding temperatures

As we can see, for low temperatures (T $<$1000 K) this combined effect on the specific isobaric heat capacity and on the gas density has a great impact on the evaporation times, while for temperatures higher than 1000 K the effect fades and the time difference is less pronounced: the lines shown in the graph almost tend to converge.

Therefore, we can say that for temperatures below 1000 K the surrounding pressure has a greater effect on the evaporation time of the particles, exceeding the 1000 K we have a greater influence given by the temperature, with the pressure differences that become almost less pronounced.

Another interesting thing, not considered in our study, which affects tangibly the evaporation time of the particle due to the surrounding pressure, is that pressure changes the saturation liquid mole fraction at the particle's surface, because of their relation described by the *Clausius-Clapeyron equation*. A detailed explanation of this relationship and how it affects the evaporation time of the particle is reported in the study of By Shashank, E Knudsen and H Pitsch [22].

In this case we've added also how the liquid penetration (the distance traveled by the liquid particle before evaporating) changes due to the variation of the surrounding pressures:

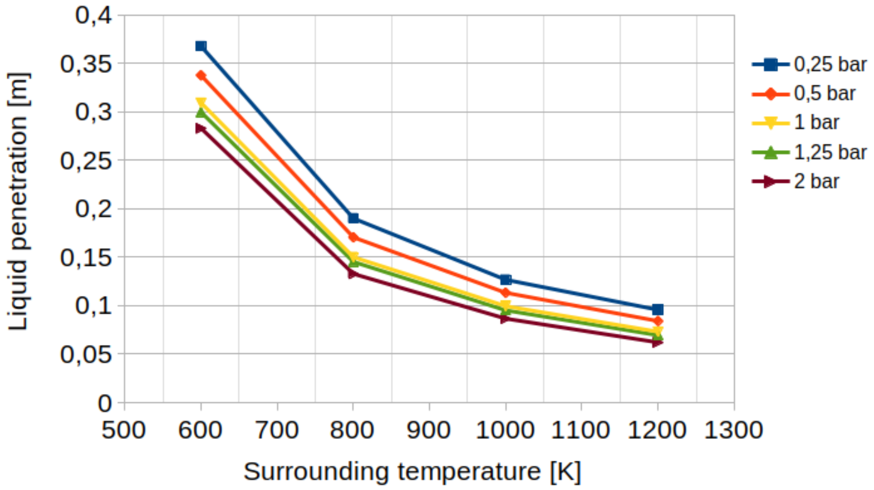


Figure 16: Liquid penetration for different surrounding pressures and with different surrounding temperatures

What we can see from the graph 16 is quite simple to explain: with an higher environmental pressure it's obvious that the liquid penetration will be shorter: the lifetime of the liquid particle is less and even the higher pressure causes a greater density of the gas, which increases the viscosity of the gas, opposing the advancement of the liquid particle.

We've that the viscosity is defined, informally, as the fluid's resistance to flow, as defined by the French physiologist *Jean Poiseuille*.

It's possible to see all the different values of liquid penetration from Table 6:

T_a [K]	0,25 bar	0,5 bar	1 bar	1,25 bar	2 bar
600	0,37	0,34	0,31	0,3	0,28
800	0,19	0,17	0,15	0,14	0,13
1000	0,13	0,11	0,1	0,1	0,09
1200	0,1	0,08	0,07	0,07	0,06

Table 6: Liquid penetration data for different surrounding pressures

5.6 How evaporation time changes by varying the characteristics of the mesh

Now we show how the evaporation time changes by varying the characteristics of the mesh, as size or grading, in order to choose the best configuration, paying attention to both the accuracy of the data obtained and the calculation time required.

We have to pay attention to the duration time due to the Courant number (explained before in sub-chapter 4.4) because we've decided to use a certain value for the Courant number (around 0.2 in almost all cases, but later also the value of Co = 0.3 will reveal to be useful). OpenFOAM requires the specification of the maximum Courant number allowed, and it'll try to mantain it behind this value, by changing the value of the time step.

5.6.1 Changing the size of the Mesh in all directions

The initial configuration of the mesh is a set of cubes with a side of 1 cm, for a total volume of each cell of 1 cm^3.

Changing the values seen before (number of cells in each direction), by increasing (2 cm; 4 cm) or decreasing it (0,5 cm; 1 mm), we've collected the data for two surrounding temperatures, in order to analyze how much the evaporation times are changed:

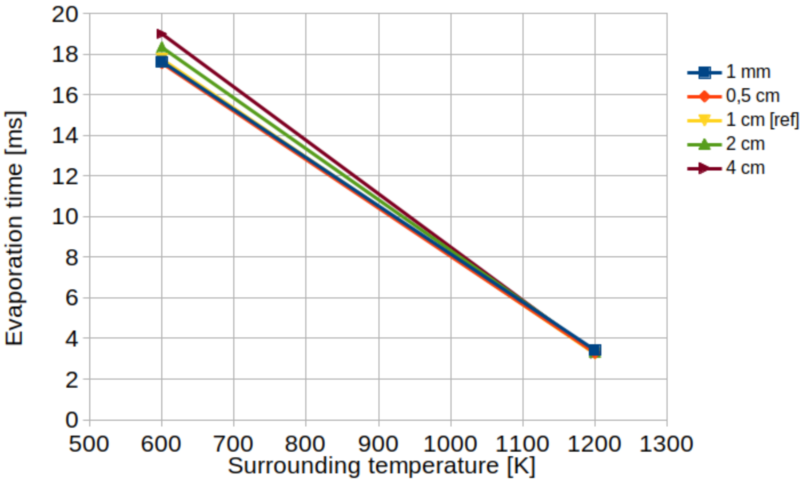


Figure 17: Evaporation time data for different cells' sizes at two reference temperatures

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	17,63	17,56	17,75	18,33	19
1200	3,42	3,29	3,25	3,33	3,25

Table 7: Evaporation time data for different cells' sizes at two reference temperatures

Also the data of the liquid penetration of the particle are changed, as we can see from Fig. 18 and Table 8:

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	0,343	0,308	0,308	0,310	0,312
1200	0,082	0,075	0,073	0,070	0,064

Table 8: Liquid penetration data for different cells' sizes at two reference temperatures

In the end we can conclude, that with regard to the evaporation time the differences between a mesh of 1 *mm* and that of 1 *cm* can also be neglected.

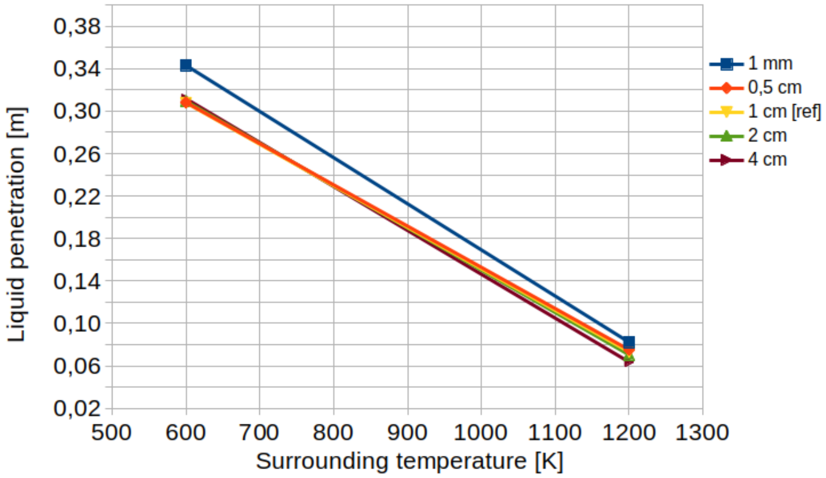


Figure 18: Liquid penetration data for different cells' sizes at two reference temperatures

The situation changes with regard to the liquid penetration distance, which will particularly interest us when we move on to the study of liquid spray (section 6).

The fact that we see more marked differences in the liquid penetration length with respect to the evaporation time, suggests that OpenFOAM, to keep the Courant number within the set limit ($Co_{max} = 0.3$), has changed more the value of the time step.

5.6.2 Changing the size of the Mesh in x-direction

We report on Fig 19 and on Table 9 as the evaporation time changes and liquid penetration data on Fig 20 and on Table 10, obatined just by changing the side of the cells in the x-direction: we won't have cubic cells, but rectangular ones.

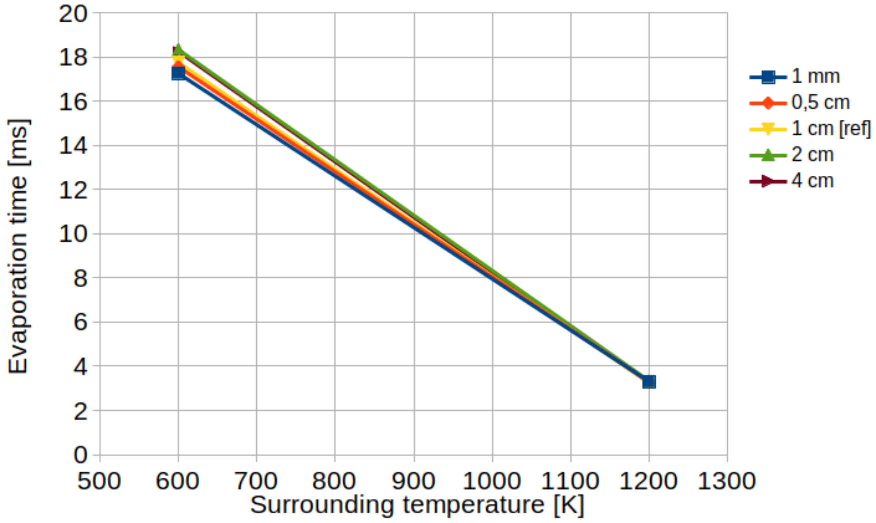


Figure 19: Evaporation time for x different cells' sizes at two reference temperatures

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	17,26	17,56	17,75	18,33	18,25
1200	3,31	3,29	3,25	3,33	3,25

Table 9: Evaporation time data for different x cells' sizes at two reference temperatures

While the liquid penetration data are:

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	0,312	0,308	0,308	0,307	0,307
1200	0,074	0,073	0,073	0,073	0,072

Table 10: Liquid penetration data for different x cells' sizes at two reference temperatures

We can notice some slight differences with respect to the finished mesh in each direction only for the case in which we have adopted a length along

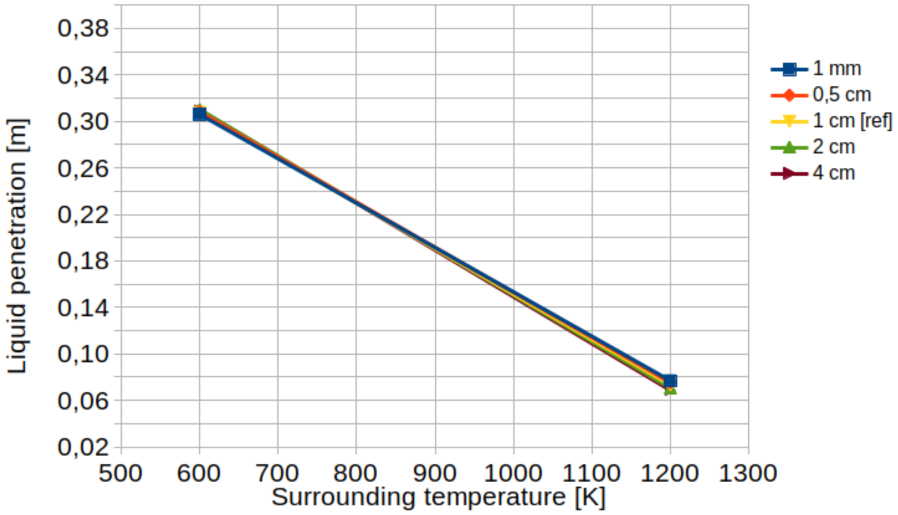


Figure 20: Liquid penetration data for different x cells' sizes at two reference temperatures

the x direction of a millimeter, so we find it easier, at least from the results obtained so far, to adopt a mesh of one millimeter but only in the x direction of movement x.

5.6.3 Changing the size of the Mesh in y-direction and z-direction

To notice the differences compared to the previously tested configuration, we tried to change the fineness of the mesh also along the other two directions, not the movement ones, to test if they have an impact on the results.

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	17,78	17,75	17,75	17,75	17,83
1200	3,22	3,25	3,25	3,25	3,25

Table 11: Evaporation time data for different y and z cells' sizes at two reference temperatures

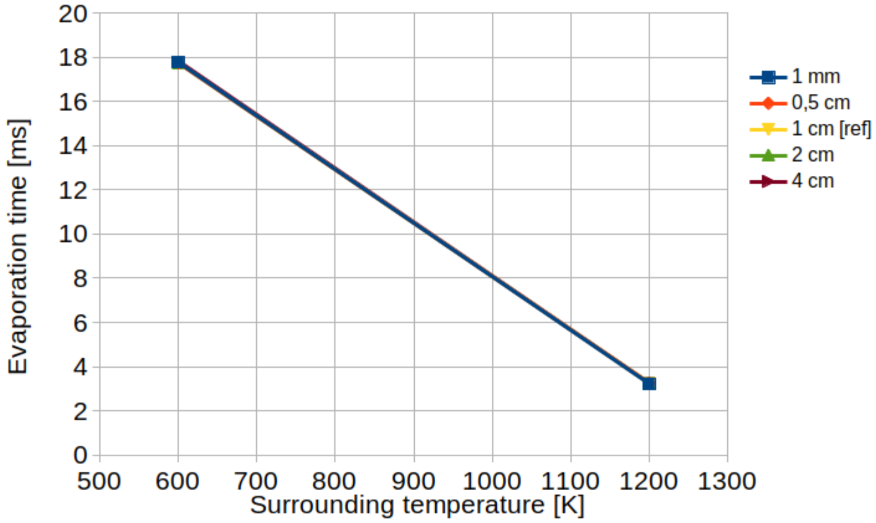


Figure 21: Evaportaion time data for different y and z cells' sizes at two reference temperatures

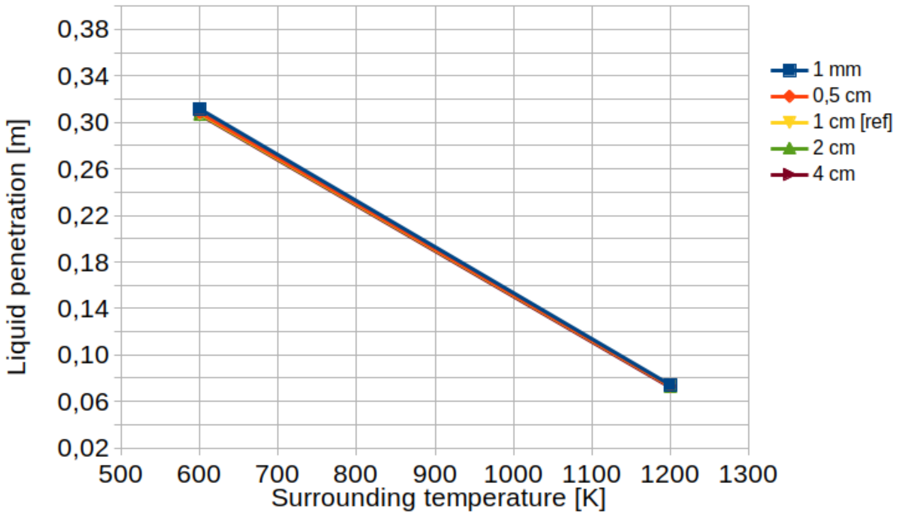


Figure 22: Liquid penetration data for different y and z cells' sizes at two reference temperatures

T_a [K]	1 mm	0,5 cm	1 cm [ref]	2 cm	4 cm
600	0,312	0,308	0,308	0,307	0,307
1200	0,074	0,073	0,073	0,073	0,072

Table 12: Liquid penetration data for different y and z cells' sizes at two reference temperatures

We can therefore conclude from the data reported in the tables and in the graphs shown, that the use of finer cells is helpful in the moving direction of the particle (x-direction), while in the other two directions (y-direction and z-direction), transversal to the first one, the relative error on the calculation of the evaporation time and of the liquid penetration can be considered negligible.

5.6.4 Relative Errors and the best Mesh settings

After reporting the results of the simulations with different mesh sizes, it's possible to compute the absolute error, by setting a reference case (that one with a mesh size of 1 cm), in order to compute the relative error. Once computed the relative error, the best mesh settings can be identified, or it'll be possible to see when the error could be considered as neglegible.

The **relative error** represents the instrumental imperfection through which the measurement is taken. It's given by the ratio of the difference between the measured value and the exact value itself (the reference case, in the present work). The relative error is calculated as follows:

$$E_r = \frac{\tau - \tau_{ref}}{\tau_{ref}} \tag{78}$$

where τ_{ref} is the the reference value while τ is the value obtained after changing the mesh settings.

Through this equation, the relative errors made by changing the mesh size with respect to the reference case, are obtained. They are reported in the two tables below, for the evaporation time and the liquid penetration:

Ta [K]	xyz [%]	x [%]	yz [%]
600	0,64	2,78	0,15
1200	5,35	1,78	0,85

Table 13: Relative error in percentage on the evaporation time with different mesh sizes

While, for the liquid penetration, the data are:

Ta [K]	xyz [%]	x [%]	yz [%]
600	11,45	0,61	1,19
1200	12,26	5,01	1,28

Table 14: Relative error in percentage on the liquid penetration with different mesh sizes

From the data above, it's easy to see how much the moving direction (x-axis) reveals to be the most important one.

It is interesting to note that the relative error appears to be grater for the liquid penetration: probably the variation of the mesh size.

The results show how much important is the mesh refinement and that our configuration is good and that, if we wanted to save in terms of computational time, we could only modify the x-direction, while the sizes on the y-direction and z-direction need to be the same, in order to catch closer results to the real case.

It is emphasized that the penetration of the liquid changes a lot between the setting of the mesh from 1 *cm* to 1 *mm*, and for this reason, in the next case study, in which we have analyzed the liquid spray, for which the values of the penetration of liquids and vapors is very important as a comparison tool, we will change the mesh to the value of 1 *mm*.

6 CFD Analysis of the duct with the liquid spray jet

After analyzing how the initial conditions and the mesh'size influence the behavior of the single particle (we have considered only the liquid penetration and the evaporation time since being a single particle it does not come into contact with anything other than the vapor phase in the duct, so it does not give rise to any other phenomenon relevant to our study), we can move on to analyze a spray jet injected into the duct.

In moving to the liquid spray we've analyzed how it works, which are the main phenomena that affect it and how changing some initial conditions, plus others that are typical of liquid sprays (think of the phenomena of breakup, those of atomization or even to the turbulent collisions between the various particles), influence our data and the behavior of the liquid phase in contact with the gas phase.

6.1 Our model configuration

In the picture reported in the next page we've a sketch of one section of the computation domain in which we've injected our liquid dodecane through a *pressure driven flow.*

The square duct used to study the injection of the dodecane ($C12H26$) has a length of 80 mm, and a square section of 20 mm side in the other two spatial directions. As shown in the sub-section 6.4 we decided to adopt a finer mesh and a geometrical domain in mm differently from the case of the single droplet in order to have a higher precision in studying the interaction between the injected liquid phase and the gas phase present in the duct. The geometric domain has been reduced in size to guarantee a lower computational cost, without however altering the results with respect to the configuration in cm, as shown by the comparison made on the sub-chapter 6.4.

The pressure of injection for the liquid spray is equal to 152.5 MPa (following the configuration used for the experimental results taken from Sanghoon Kook and Lyle M Pickett [23]), while the velocity of the gaseous phase along the duct (along the x-direction) is equal to 0.2 m/s.

Regarding the initial conditions, the case examined to validate our physical model has an ambient temperature of 891.9 K and a pressure of 6.05 MPa.

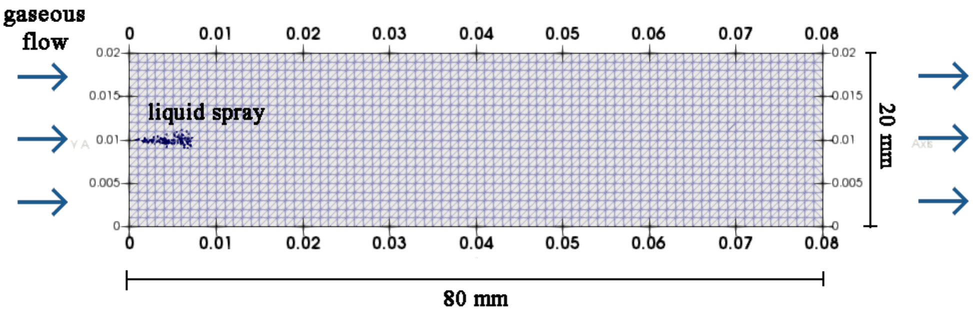


Figure 23: Graphical representation of one section of our case configuration with the geometrical dimensions and the particles injected into the duct, with the gaseous flow that runs through it.

The gaseous phase within the duct is, instead, a mixture of $CO2$ (6.52%), $N2$ (89.71%) and $H2O$ (3.77%), whose percentages refer to the mass fraction of the surrounding gas phase.

6.2 Validation of the model

In order to validate our simulation model, we decided to adopt, as liquid spray to be injected into the duct, the dodecane ($C12H26$). This choice is motivated by the availability of experimental data given us by the ECN (Engine Combustion Network), in which are reported data about the evaporation time of a liquid spray jet injected into a gaseous phase of air (experiments made by Sanghoon Kook and Lyle M Pickett [23]).

The flow of the vapor phase has a turbulent behaviour, that we've decided to study through the $k - \epsilon$ *model*, with the *RANS approach*, that we've explained in the sub-chapter 3.3.4.

In this case our comparison is based on the data of liquid and vapor penetration, which in the study carried out by Sanghoon Kook, Lyle M. Pickett [23], were obtained experimentally through the use of simultaneous high-speed imaging of Mie-scattering and schlieren at approximately 50-kHz framing rates, studying the injection made by a standard common-rail diesel injector (Bosch, second generation). Just to simulate the use of a common-rail injector, in our simulation we've used a $pressureDrivenVelocity$ flow, with a pressure of injection equal to 152.5 MPa.

In order to register the data of vapour penetration, we've placed 10 probes along the central x-axis (the main direction of motion), as we've shown in sub-chapter 6.3.1, so in our comparison we've to take into account a slight inaccuracy in the data obtained, due to the approximated position of the probes.

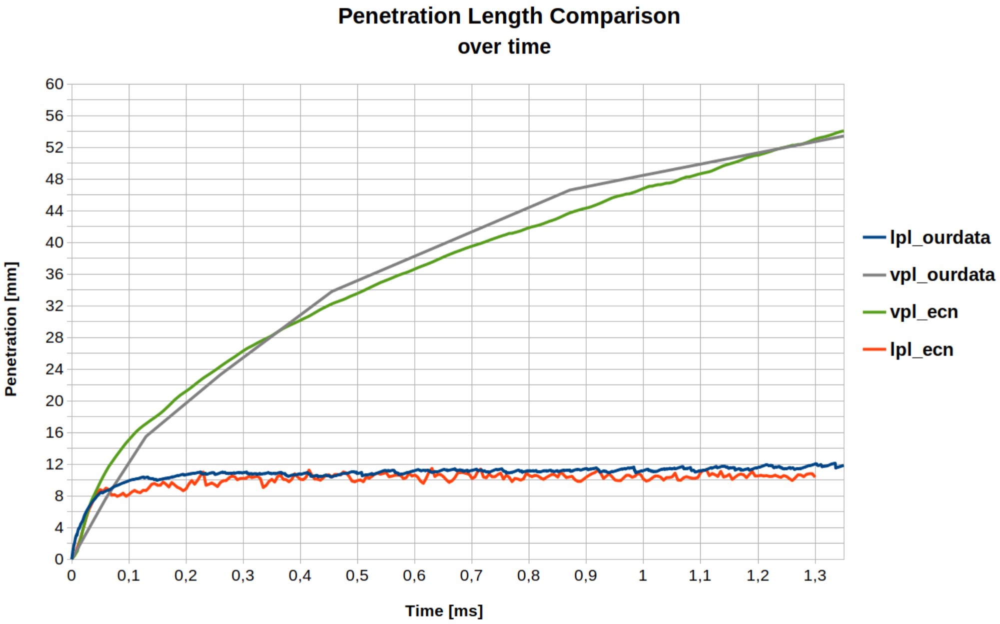


Figure 24: Results related to liquid penetration and vapour penetration compared with experimental ones obtained by Sanghoon Kook, Lyle M. Pickett [23]

With the indexes that refer to the case from which the data are obtained: the experimental data present the "ecn" index, while the model that we have validated by comparing ourselves to these data has the "our" index.

Time [ms]	LPL_our [mm]	LPL_ecn [mm]	VPL_our [mm]	VPL_ecn [mm]
0,06	8,62	8,97	7,71	11,09
0,13	10,33	8,65	15,52	17,11
0,26	10,91	9,69	23,33	24,34
0,46	10,41	10,37	33,84	32,24
0,87	11,27	10,39	46,62	43,69
0,99	11,13	10,67	48,42	46,67
1,97	-	-	62,21	65,24

Table 15: Data related to liquid penetration and vapour penetration compared with experimental ones obtained by Sanghoon Kook, Lyle M. Pickett [23]

6.3 Penetration lengths

Also the data reported on the table on the previous page confirm that we've some little inaccuracies related to the positions of the probes along the duct, that are able to catch the percentage of the gas phase within a certain cell just for some time steps in certain positions, and not for every single millimeter.

For the measurement of the Liquid Penetration Length (LPL) and the Vapour Penetration Length (LPL) we've considered the usual definitions of them accepted in CFD studies:

- Liquid Penetration Length: the distance traveled axially from 95 % of the mass fraction of the liquid phase from the exit of the nozzle;
- Vapour Penetration Length: the distance traveled axially from 0.1 % of the mass fraction of the vapour phase, originated by the evaporation of the liquid phase injected, from the exit of the nozzle.

We've used the 95 % of the mass fraction of the liquid phase for the definition of the Liquid Penetration Length, because in OpenFOAM at the injection the software injects also a little percentage of vapour phase, that approximatly is equal to the 5 % of the mass fraction.

And the results obtained from the measurements of the Vapour Penetration Length through the use of probes placed along the x-axis of the duct could be also compared with the results that we can obtain from a "mechanical" estimation at the post-processing part with the use of the "ruler" function, as it's possible to see on Fig. 25 on the next page.

About the values of the VPL is easy to understand why the gas does not expand along the entire duct: it has a pressure of injection greater than the environmental pressure, but only while it is in the liquid phase (for this reason we've good values of liquid penetration length), but after the phase change we've that the high value of the surrounding pressure stops the expansion of the evaporated dodecane due to a higher value of density of the surrouding gas (Eq. 18).

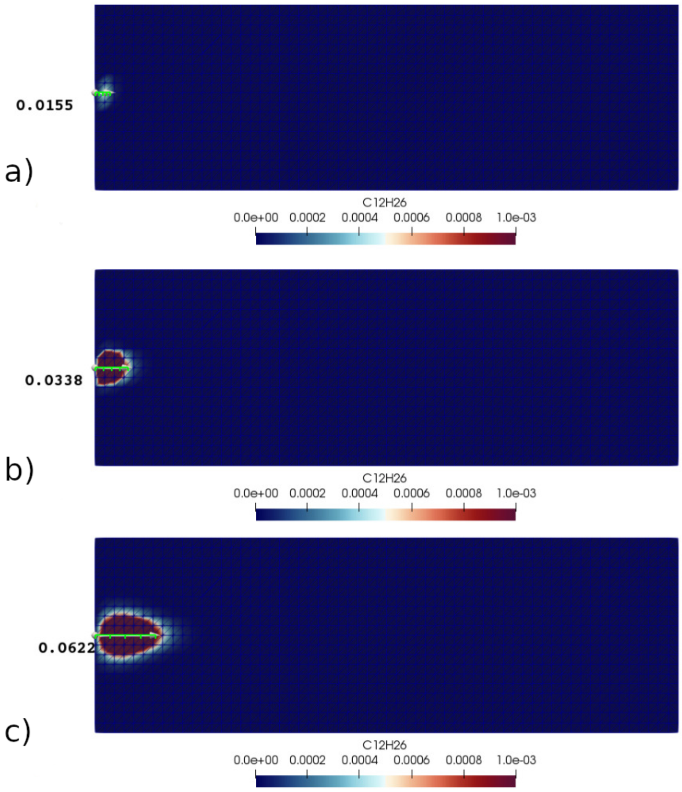


Figure 25: Vapour Penetration Length in three different time steps measured on Paraview through the "ruler" function, to make a comparison with the results given by the probes.

a) t= 0,13 ms; VPL= 15,5 mm
b) t= 0,455 ms; VPL= 33,8 mm
c) t= 1,97 ms; VPL= 62,2 mm

6.3.1 Procedure to measure the vapour penetration length

One of the main challenge that we've met during our study of the model is how to get the vapour penetration lengths.

While the liquid penetration lengths are automatically measured by sprayFoam, thanks to the Lagrangian Paricle Tracking (LPT) approach, that follows the liquid particles during their path, measuring their spatial position at each time step.

The research on how to measure the VPL has taken a long time, as there was a need to conceive conceptually the definition of length of vapor penetration to be able to apply it to the Eulerian study of the gas phase, which has as unique points of spatial reference the nodes that constitute each cell of the mesh.

In the end we came to the conclusion that the only way to get the vapour penetration length was to place "probes" along the x-axis and place each probe at the same distance from the next. Therefore, we placed ten probes along the x axis: not along its entire length, but right along a segment of the same length as the maximum length of vapor penetration obtained from the experimental data (more and less 80 *mm*).

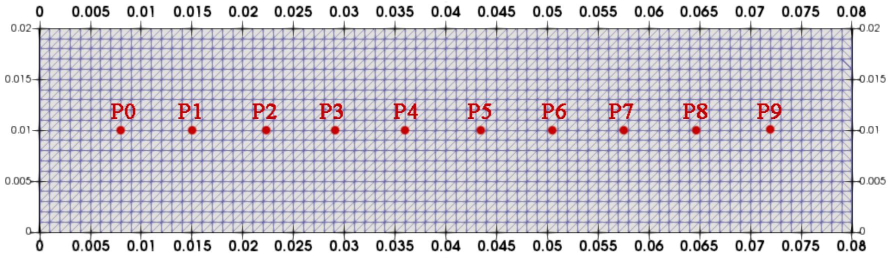


Figure 26: A section of our computational domain with the ten probes placed along the x-axis, in the middle of the duct.

Probe	P0	P1	P2	P3	P4	P5	P6	P7	P8	P9
xp [mm]	8	15,1	22,2	29,3	36,4	43,5	50,6	57,7	64,8	71,9

Table 16: Positions of the ten probes placed along the x-axis.

6.4 Geometric field and mesh size

In order to have a post-processing part more detailed without increasing too much the computational cost, we've adopted a geometrical configuration 10 times smaller (passing from cm to mm, differently from the single droplet case), but without changing the parameters of our model for the liquid spray. This choice is corroborated by the fact that we will be able to exploit a finer mesh, which in addition to a better visualization in the post processing, will make our data slightly more accurate, having reduced the distance between the various nodes of the grid.

However, it's necessary to consider that our results will be a bit different, because of the dimension of the duct that will be 10 times smaller than the original domain chosen (in *cm*): this will bring some effects to some fluid dynamic properties (e.g. Reynolds Number *Re*).

This one below is a comparison with the work described by Sanghoon Kook and Lyle M. Pickett [23] with our reference model both in cm and in mm.

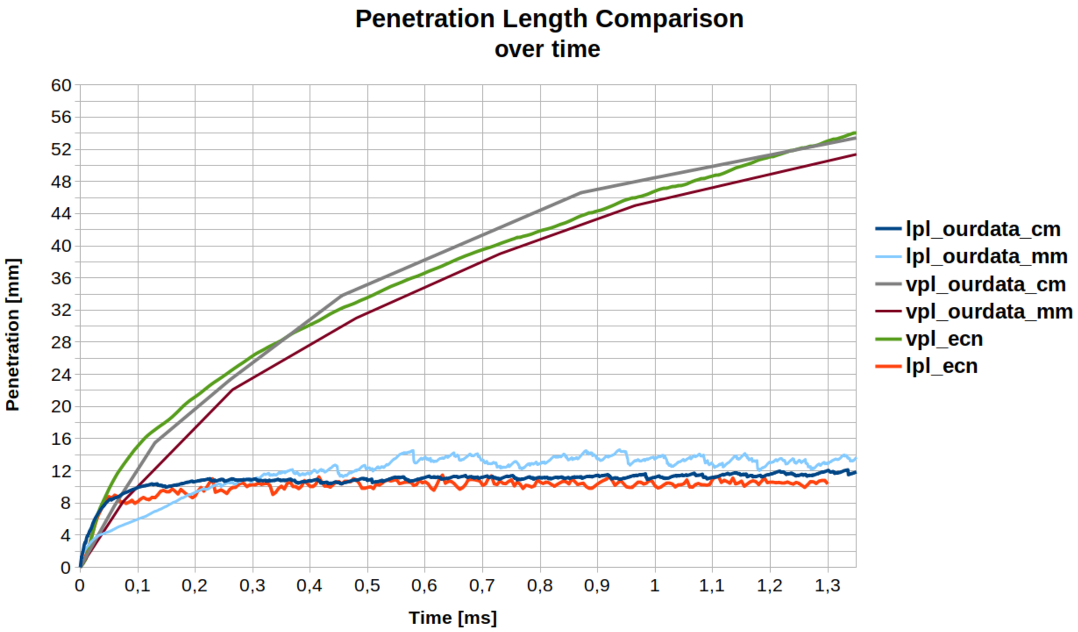


Figure 27: Comparison between the reference model of Sanghoon Kook and Lyle M Pickett [23], our model in cm (as initially chosen) and our own model in mm.

Looking at the data reported by the graphs above, we can conclude that the change from cm to mm does not significantly change our results, and that the savings in terms of computational cost justify these slight inaccuracies.

There is also to consider that the deterioration affects only the liquid penetration, while that of steam is improved, given that it is based on a finer and therefore more precise Eulerian grid.

The length of liquid penetration, on the other hand, based on the Lagrangian method, does not depend on the mesh: we will have a "duration" of the greater liquid particles, probably because a smaller duct allows a turbulence of lower intensity, therefore with less heat exchange between the two phases, and therefore with a greater duration of the particles.

6.5 Phase change

About the phase change model chosen for our simulation, we've decided to adopt the **liquidEvaporationBoil** model: our particles reach boiling temperature and evaporate gradually, not instantaneously. In OpenFOAM are available two models: the *liquidEvaporationBoil* and the *liquidEvaporation*; in the second one the particles evaporates instantaneously after reached the temperature of evaporation, without carrying out a gradual phase shift and distributed over time: in that case we've a smaller lifetime of the particles and a situation that is simpler to be studied.

We can see from the "graphical" comparison with the work described by Sanghoon Kook and Lyle M. Pickett [23] with our work (Fig. **??**), that effectively the values of Vapour Penetration Length (VPL) and Liquid Penetration Length (LPT) are very similar and they occur at the same time steps. The VPL shown on the pictures, however, is set to a percentage value of dodecane equal to 2.5%, because the gas phase percentage values that can be identified by the experimental equipment used in the work by Sanghoon Kook and Lyle M. Pickett [23] isn't specified.

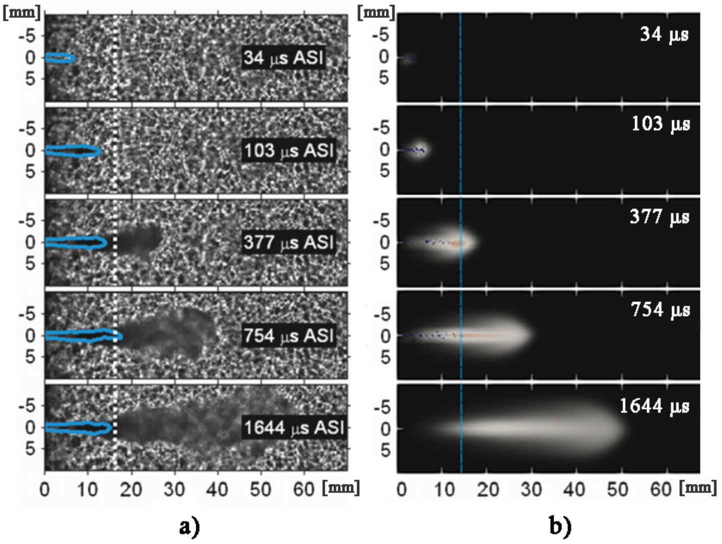


Figure 28: Graphical comparison between (a) the row schlieren images by [23] and (b) our simulation

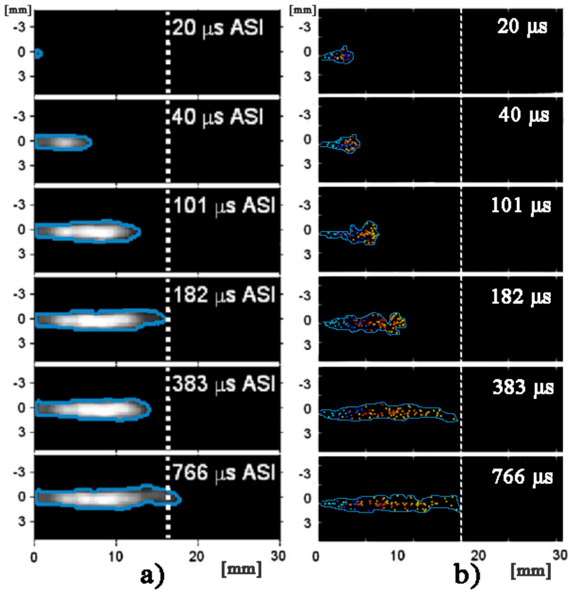


Figure 29: Graphical comparison of the liquid penetration length (LPT) between the images (a) by [23] using the Mie-scattering method and (b) our simulation

6.6 Breakup models comparison

In the sub-chapter 3.10 about the theoretical description of which phenomena occur at the spray injection, there's the theoretical description of the secondary breakup model used in this simulation, to explain how it works.

The Kelvin-Helmotz Rayleigh-Taylor (KHRT) breakup model represents our final choice, because in situations like the one described in this work, in which sprays have a very high initial velocity (due to the high injection pressure) and decelerate rapidly (due to the very high density of the gaseous phase, because of the high pressure), turns out to be the best model thanks to the use of the Rayleigh-Taylor instabilities. This assumption is also justified by the study carried out by S Hossainpour and A R Binesh[5] and also reinforced in the Siemens' User guide [9].

Despite this, in an attempt to carry out a study as accurate as possible, it was decided to make a comparison with the results obtained using the breakup model that comes closest to our experimental data, in addition to the KHRT model already used, as it's possible to see in the figure below:

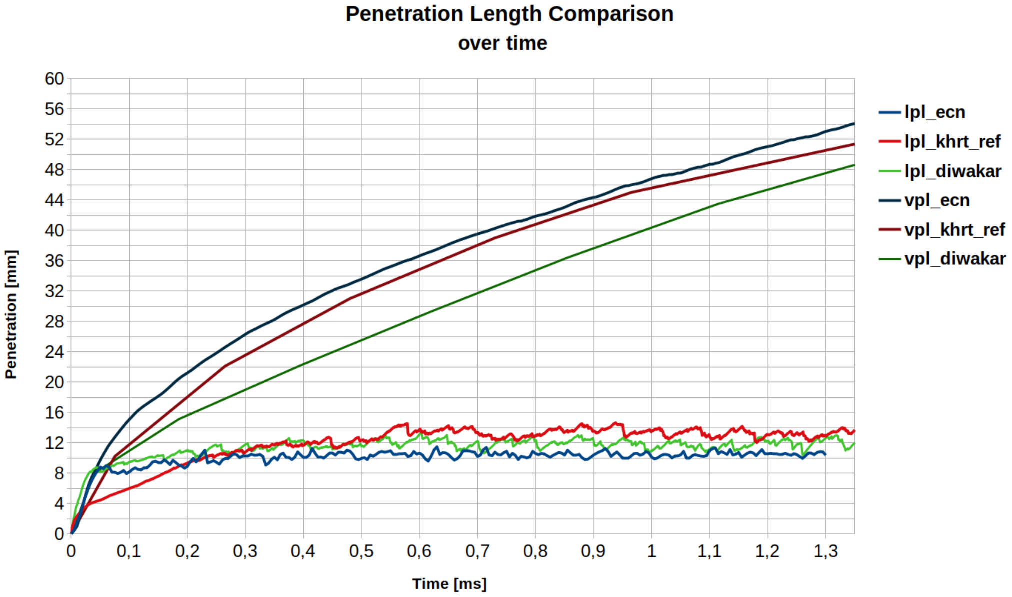


Figure 30: Comparison of the results obtained by changing the secondary breakup model.

Time [ms]	LPL_KHRT [mm]	LPL_ReitzD [mm]	LPL_ECN [mm]
0,08	5,26	9,05	8,15
0,27	10,48	10,78	9,92
0,48	11,99	11,90	10,89
0,73	12,60	11,90	10,76
0,97	13,29	12,18	10,27

Table 17: Data of Liquid Penetration Length plotted on Fig. 30

Time [ms]	VPL_KHRT [mm]	VPL_ReitzD [mm]	VPL_ECN [mm]
0,08	10,20	9,20	12,74
0,27	22,10	18,10	24,59
0,48	31,00	25,30	32,93
0,73	39,00	34,00	40,26
0,97	45,00	40,80	46,00

Table 18: Data of Vapour Penetration Length plotted on Fig. 30

From the data reported in the tables above, it's possible to see that the KHRT breakup model reproduces more precisely the penetration of vapor, while slightly overestimates, compared to the breakup model ReitzDiwakar, the liquid penetration. Calculating the relative error with reference experimental data (provided by the ECN) on vapour penetration and liquid penetration of both breakup models, it's possible to see that the KHRT breakup model results effectively to be better than the ReitzDiwakar model:

LPL Relative Error [-]			
Time [ms]	Er_KHRT	Er_ReitzD	OPTIMAL
0,08	35,44	11,10	ReitzD
0,27	5,61	8,62	**KHRT**
0,48	10,04	9,28	ReitzD
0,73	17,08	10,59	ReitzD
0,97	29,48	18,61	ReitzD

Table 19: Liquid Penetration Length comparison between the KHRT breakup model and the ReitzDiwakar one.

VPL Relative Error [-]			
Time [ms]	Er_KHRT	Er_ReitzD	OPTIMAL
0,08	19,96	27,80	**KHRT**
0,27	10,13	26,40	**KHRT**
0,48	5,87	23,18	**KHRT**
0,73	3,14	15,56	**KHRT**
0,97	2,17	11,30	**KHRT**

Table 20: Vapour Penetration Length comparison between the KHRT breakup model and the ReitzDiwakar one.

The behaviour of the two models is surely different, and probably the best way to study how they work is watching to how they "divide" the particles, by observing their Particle Size Distribution (PSD) for several time steps:

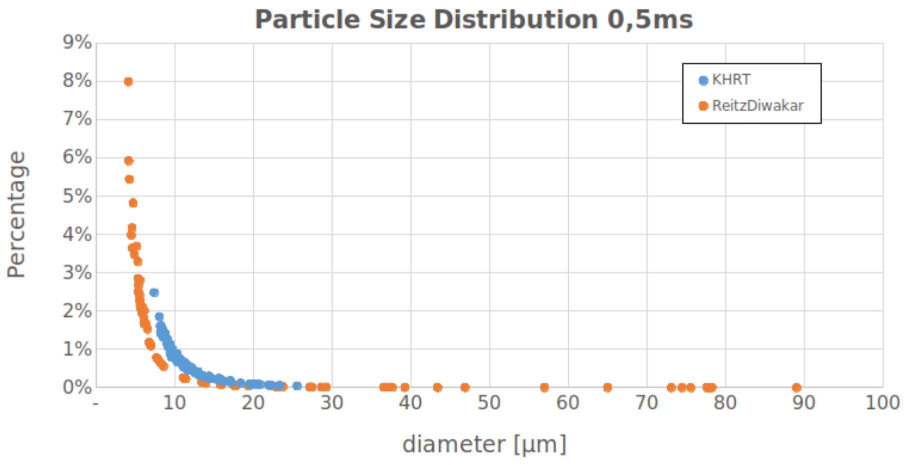


Figure 31: Paricle size distribution for the KHRT breakup model and the ReitzDiwakar one at t= 0,5ms

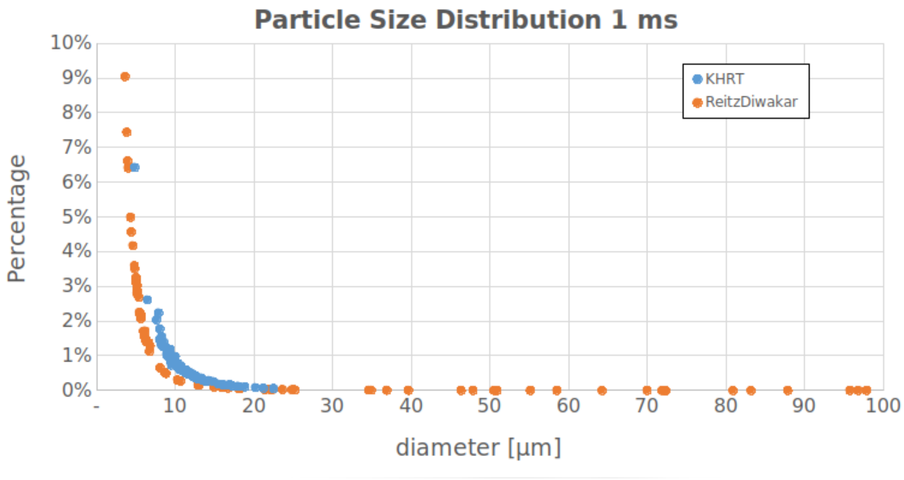


Figure 32: Paricle size distribution for the KHRT breakup model and the ReitzDiwakar one at t= 1ms

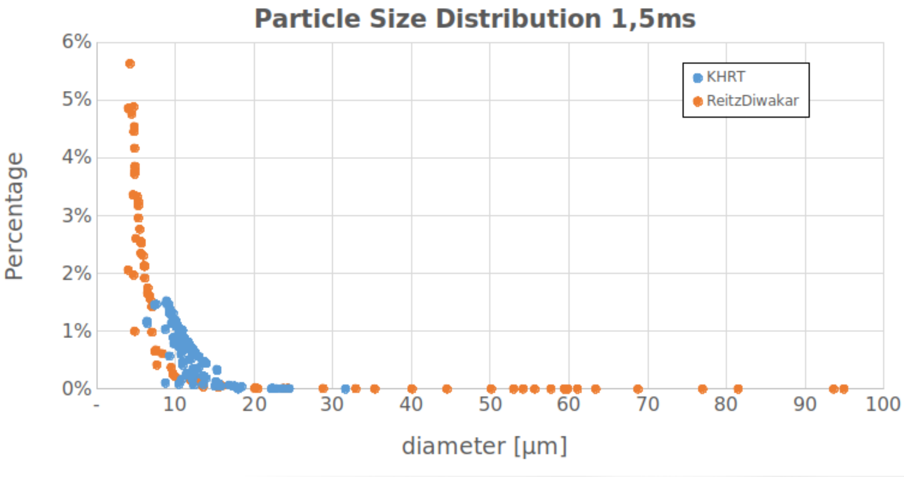


Figure 33: Paricle size distribution for the KHRT breakup model and the ReitzDiwakar one at t= 1,5ms

In the three pictures representing the Particle Size Distribution of the two breakup models, we've removed, to have a more precise graphical representation of the physical quantities, the particles with diameters lower than 1 μm, because they're very low in percentage (lower than 0,02 %) and they can be considered negligible.

Mean diameter [μm]		
Time [ms]	KHRT	ReitzDiwakar
0,5	10,37	5,39
1	9,97	5,03
1,5	10,72	5,29

Table 21: Mean diameter of the two breakup models for each time step represented in the previous pictures.

From the PSD of three different time steps, that take into account the injection duration ($1,05 ms$), and from the data reported in the table above, it's possible to see how the ReitzDiwakar breakup model, differently from the KHRT one, has a more uniform distribution of the liquid for particles of different sizes: the number of particles is smaller, but we've more particles with a greater diameter then the common diameter of the particles created by the KHRT model (around $1 \mu m$).

It's possible to conclude that the liquid particle length is greater for the KHRT model because it creates particles with a higher mean diameter (more and less 10 μm) that goes through the gaseous phase of the environment, while the the vapour penetration length of the KHRT breakup model reveals to be higher because we've a greater number of small liquid particles than the ReitzDiwakar model and, so, they release the vapour phase faster due to a more marked phase change.

6.7 Turbulence effects comparison

Other interesting things that deserves comparison are the effects of turbulence on LPL and VPL, which are probably the most interesting quantities to focus our study on: thus we dedicate a sub-chapter to the comparison between turbulent and laminar flow. A summary of the results is presented in Fig 34:

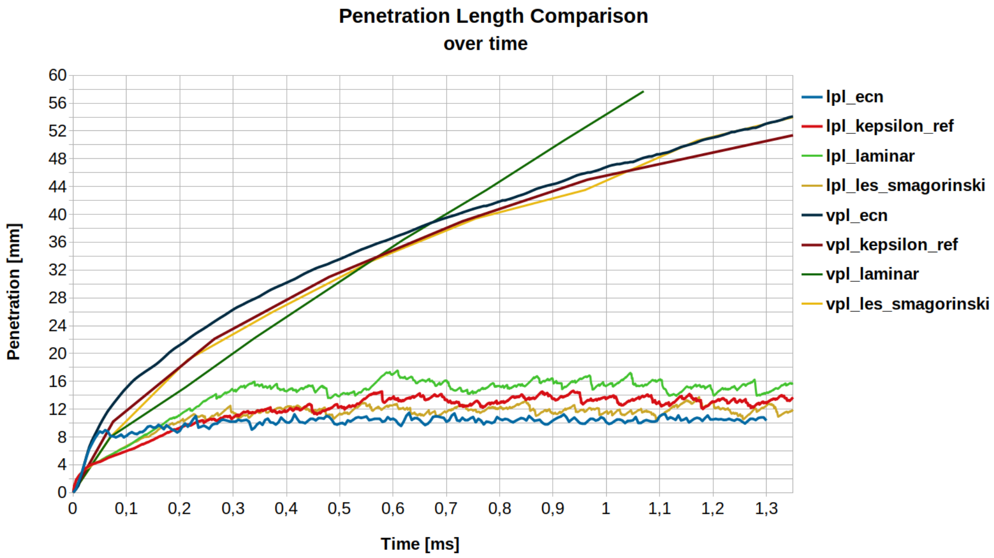


Figure 34: Comparison of the results obtained by changing the turbulence models.

6.7.1 Turbulence on/off

From the data traced in the previous image it's possible to see how the presence of a turbulent regime has an important influence on liquid penetration length and on vapour penetration length.

The LPL results are greater for the laminar flow (b) due to the absence of vortices in the gaseous phase, that tend to stop the liquid particles flow, probably because brings the particles to have a less linear motion, bending them slightly, and thus opposing their axial advancement in the direction in which they are injected: with a linear regime they are free to move in the axial direction and the drag force due to the gaseos flow pushes them in the axial direction.

The same happens for VPL: initially it is smaller than in turbulent conditions, because the particles tend to evaporate more slowly, but later, given the absence of vortices, the progress of the gas phase is also linear and therefore tends to be greater with respect to the VPL in turbulent regime.

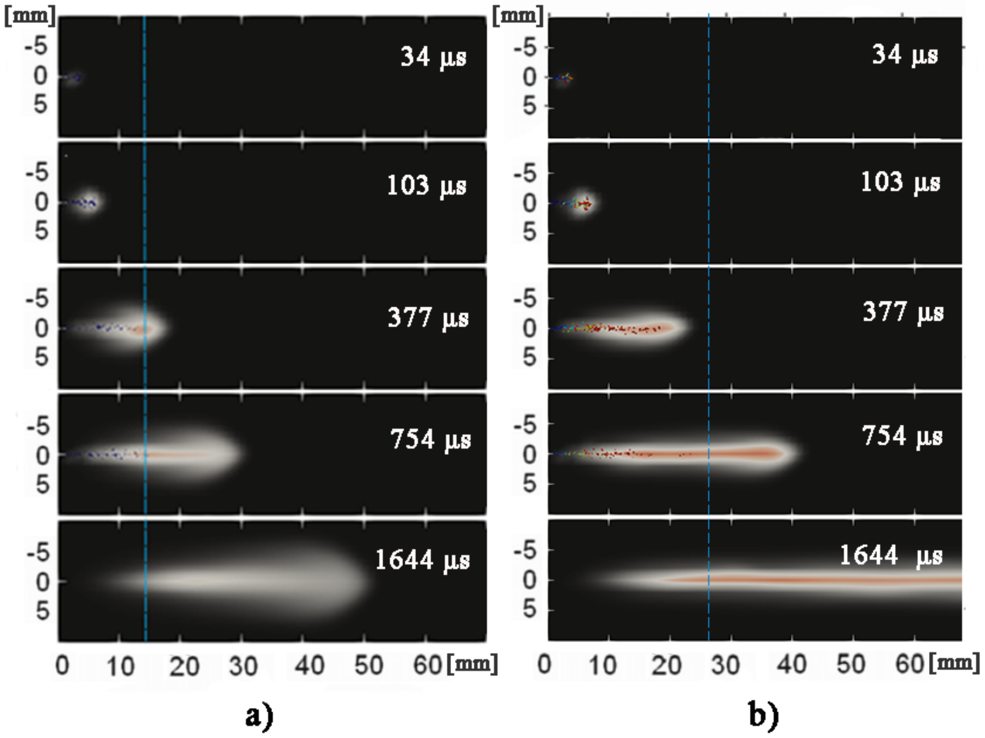


Figure 35: Graphical comparison of the results obtained by switching off the turbulence of the gaseous phase: on the left the turbulent regime (a) and on the right the laminar one (b).

6.7.2 LES turbulence model

Simply as a matter of convenience, we also reported the graph represented by the values obtained by changing the RANS turbulence model used in our reference case, with a LES turbulence model.

The use of a LES model, in which we used the Smagorinski model, probably the most used in the case of multi-phase systems, is certainly one of the possible works to be developed in the future on the spray model tested in the following work.

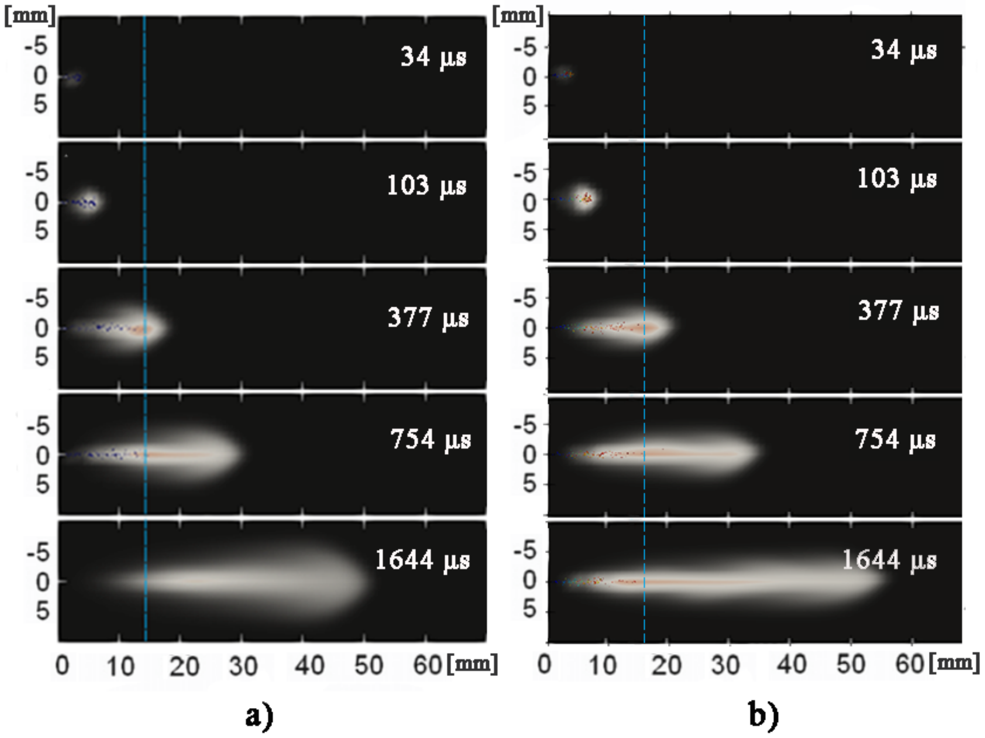


Figure 36: Graphical comparison of the results obtained by changin the turbulence model RANS turbulence model on the left (a) and the LES turbulence model on the right (b).

We will not go any further into the study, since the LES typology would require a separate section to be explained from a theoretical point of view, so we only reported the values to show how, for vpl and lpl, it approximates to the experimental data is even more precise.

7 Conclusions

At the end of our simulation work, we can conclude that:

- OpenFOAM, despite being a free software and not used in a commercial environment, is a valid and powerful tool to study the interaction between a liquid phase, in the form of particles, and a gas phase;
- the sprayFoam solver, which provides the use of a combined Eulerian-Lagrangian study, probably represents the best solver to study a multiphase flow for our spray study, also having a function to simulate the combustion reactions (not used during this study);
- the use of a finer mesh size significantly influences the Eulerian study, while the Lagrangian study is influenced indirectly and in a smaller way, since the latter takes the particle itself as a reference system, so it does not interact with the grid to study quantities as the liquid penetration length;
- changing the initial conditions, such as the pressure of the surrounding environment, the size of the liquid particle or the speed of flow between the two phases, results to be in agreement with what is established by the theoretical equations for a single droplet;
- the breakup phenomenon acts in an evident manner on the interaction between the liquid and the gaseous phase, it is therefore appropriate to choose the model of study that is most able to approximate the physical conditions that we want to simulate, in our case the Kelvin-Helmotz Rayleigh-Taylor (KHRT) method, that reproduces in a good way the aerodynamic instabilities and the effects of the interaction with the other phase that decelerates the particles;
- the turbulent phenomena are reproduced quite reliably on OpenFOAM: it is important to establish whether there is a turbulent regime or a laminar regime, and in case of a turbulent regime, the right model to approximate our system behaviour;

7.1 Future improvements

This work represents an introduction work on the interaction between liquid spray and gaseous phase, without considering many specific things: we want to specify which aspects could be developed and improved to underline other interesting aspects.

Future work could be done in every aspect not taken into account in this work:

- Implementation and analysis of the combustion process;
- Use of approaches other than the Eulerian with the LPT method one and comparison of the others with it;
- Use of a dynamic mesh (e.g. a piston) to simulate the pressure flow which opposes the injection to study its effects;
- Use of an appropriate geometric domain (such as the cylindrical shape adopted by the combustion chambers)
- Implementation of primary breakup models.

References

[1] Alireza Ghasemi. Numerical simulation of non-reactive evaporative diesel sprays using openfoam. *Master thesis, Politecnico di Milano, Energy Department*, 2017.

[2] The OpenFOAM Foundation Manual. *OpenFOAM User Guide*, 6 edition, July 2018.

[3] Likun Ma; Dirk Roekaerts. Numerical study of the multi-flame structure in spray combustion. *Delft University of Technology, Department of Process and Energy*, 2016.

[4] Andreas Höfler. Numerische modellierung des wärme-und stofftransports von einzelpartikeln aus wässriger harnstofflösung. *Diplomarbeit , Universität Karlsruhe , Institut für Thermische Strömungsmaschinen*, 2009.

[5] A.R. Binesh S. Hossainpour. Investigation of fuel spray atomization in a di heavy-duty diesel engine and comparison of various spray breakup models. *Department of Mechanical Engineering, Sahand University of Technology.*

[6] P. Ghadimi; M. Yousefifard; H. Nowruzi. Applying different strategies within openfoam to investigate the effects of breakup and collision model on the spray and in-cylinder gas mixture attribute. *Amirkabir University of Technology, Department of Marine Technology,*, May 2016.

[7] Alejandro Municio Aránguez. Non-reacting fuel spray simulation under direct injection diesel engine conditions. *Master thesis, Lund University (Sweden) / Universidad Politécnica de Madrid (Spain), Division of Fluid Mechanics*, 2012.

[8] Ansys. Ansys user manual. *Manual, Chalmers University of Technology*, 2015.

[9] Siemens. Siemens methodology manual. *Manual, Siemens Product Lifecycle Management Inc.*, 2017.

[10] Ole Jørgen Nydal Timur Bikmukhametov, Reidar Kristoffersen. Cfd simulations of multiphase flows with particles. *Master thesis, Norwegian University of Science and Technology, Department of Energy and Process Engineering*, June 2016.

[11] Stephen R. Turns. An introduction to combustion: Concepts and applications. *McGraw-Hill*, 1996.

[12] G. Rottenkolber R. Schmehl D. Giebert O. Schäfer R. Koch S. Wittig M. Burger, G. Klose. A combined eulerian and lagrangian method for prediction of evaporating sprays. *Universität Karlsruhe , Institut für Thermische Strömungsmaschinen*, 2014.

[13] Sommerfeld M. Crowe, C. and Y. Tsuji. Multiphase flows with droplets and particles. *CRC Press LLC*, 1998.

[14] P. A. Niklas Nordin. *Complex Chemistry Modeling of Diesel Spray Combustion.* PhD thesis.

[15] Fabian Peng Kärrholm. *Numerical Modelling of Diesel Spray Injection, Turbulence Interaction and Combustion.* PhD thesis.

[16] Anne Kösters; Anders Karlsson. Modeling of diesel fuel spray formation in openfoam. *Combustion Engines Reasearch Center, CFD Simulation of I.C. engines using OpenFOAM*, August 2011.

[17] Jundi Xu; Mahish Mohan; Ebrahim Ghahramani; Hakan Nilsson. Modification of stochastic model in lagrangian tracking method. *Chalmers University of Technology*, 2016.

[18] Gaurav Tomar1 Mohit Jain, R.SuryaPrakash and R.V. Ravikrishna1. Secondary breakup of a drop atmoderate weber numbers. *Department of Mechanical Engineering, and2Department ofAerospace Engineering, Indian Institute of Science Bangalore, Karnataka 560012, India.*

[19] Michael Schlottke. Analytical investigation of droplet breakup in accelerated flow. *Universität Stuttgart, Institut für Thermodynamik der Luft- und Raumfahrt (ITLR).*

[20] Snelling D.R. Smallwood G.J., Gülder Ö.L. The structure of the dense core region in transient diesel sprays. *Symposium (International) on Combustion, vol. 25.*

[21] Robert Kasper. Particle simulation with openfoam: Introduction, fundamentals and applications.

[22] By Shashank; E. Knudsen; H. Pitsch. Spray evaporation model sensitivities. *Center for Turbulence Research, Annual Research Briefs*, 2011.

[23] Lyle M. Pickett Sanghoon Kook. Liquid length and vapor penetration of conventional, fischer–tropsch, coal-derived, and surrogate fuel sprays at high temperature and high-pressure ambient conditions. *School of Mechanical and Manufacturing Engineering, University of New South Wales, Sydney NSW 2052, Australia.*