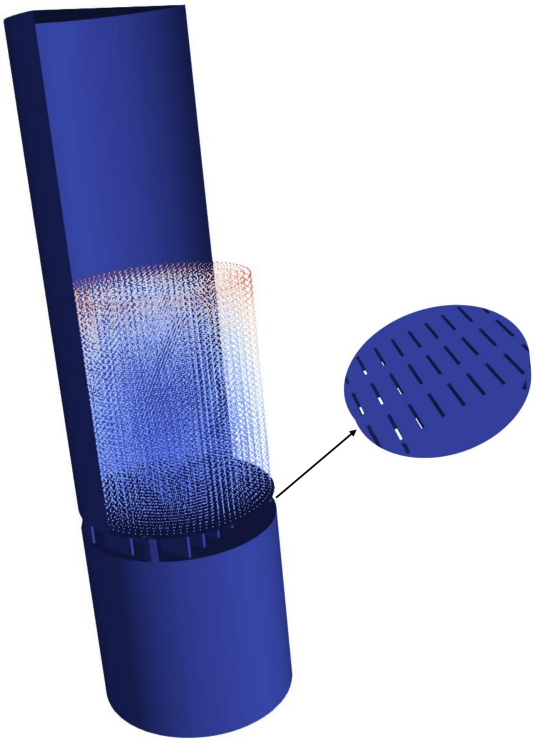
CHALMERS
UNIVERSITY OF TECHNOLOGY



Numerical modelling of pot furnace to oxidize iron ore pellets in OpenFOAM

Master's thesis in Applied Mechanics

HEMANTH KUMAR SIVASANKAR & SRIRAM SRIDHAR

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2021
www.chalmers.se

Master's thesis 2021

Numerical modelling of pot furnace to oxidize iron ore pellets in OpenFOAM

HEMANTH KUMAR SIVASANKAR & SRIRAM SRIDHAR

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
Gothenburg, Sweden 2021

Numerical modelling of pot furnace to oxidize iron ore pellets in OpenFOAM
HEMANTH KUMAR SIVASANKAR & SRIRAM SRIDHAR

© HEMANTH KUMAR SIVASANKAR & SRIRAM SRIDHAR, 2021.

Supervisor : Örjan Fjällborg, Research Engineer, LKAB

Examiner: Håkan Nilsson, Mechanics and Maritime Sciences, Division of Fluid Dynamics, Chalmers University of Technology

Master's Thesis 2021:10.
Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Pot furnace with pellets and roster plate

Typeset in LaTeX
Gothenburg, Sweden 2021

Numerical modelling of pot furnace to oxidize iron ore pellets in OpenFOAM
HEMANTH KUMAR SIVASANKAR & SRIRAM SRIDHAR
Department of Mechanics and Maritime Sciences
Chalmers University of Technology

Abstract

With the rising environmental issues within the mining industry, it is vital to go fossil-free in fuel selection while working with processes such as combustion and heating. Today fossil fuels such as coal and oil are used for heating and sintering of pellets. The green balls (iron ore concentrate) are heated to produce pellets. The production of green balls are done using bentonite as a binder in the pellet feed which is subsequently rolled. In turn, these green balls are dried, heated, and oxidized before sintering to make them suitable for the metallurgical process.

The study of pelletizing process is done using an experimental pot furnace. The pot furnace aids us with the measured data when the green balls are heated within the furnace with different gas compositions. Pellet bed samples are collected to support further analysis. This process is verified using an open-source platform that contains an oxidation model built with the platform. A solver suitable for this purpose is chosen and used in accordance to fulfill the objective of creating a numerical model of this pot furnace used in experiment. Before performing the pot furnace simulations with the chosen solver, a verification study is performed between a moving bed of pellets and a fixed bed of pellets for a smaller domain and it is found that the chosen solver is suitable for a fixed bed case. The pot furnace simulations are performed with two different activation energies obtained from literature's. It is concluded that the activation energy and oxygen supply plays a vital role in conversion process.

Keywords: OpenFOAM, reactingHeterogenousParcelFOAM, Fossil free, Pellets, Particle positioning, Activation energy, Magnetite, Hematite

PREFACE

The Master Thesis project work is carried out from January 2021 to June 2021 with LKAB remotely. The project is supervised by Örjan Fjällborg at LKAB remotely and examined by Håkan Nilsson at the Chalmers University of Technology.

ACKNOWLEDGEMENTS

We would like to thank our supervisor Örjan Fjällborg, Research engineer at LKAB, for his fruitful support throughout the tenure of the master thesis project. He was of invaluable assistance to us constantly providing inputs with his plethora of knowledge on OpenFOAM. The remote desktop servers at LKAB were especially useful resources which helped us throughout the entire course of the thesis project.

We would also like to acknowledge Håkan Nilsson, Full Professor, Mechanics and Maritime Sciences, Division of Fluid Dynamics, for playing a crucial role as not only an examiner but a supervisor within Chalmers whose inputs and knowledge on OpenFOAM have helped us reach the point where we stand. The support and guidance we received from Håkan aided us to a large extent.

We would like to extend our gratitude to Saeed Salehi, Postdoc, Fluid Dynamics/Mechanics and Maritime Sciences, and Ananda Subramani Kannan, Postdoc, Fluid Dynamics/Mechanics and Maritime Sciences, for their valuable inputs during the entire course of the master thesis.

Hemanth Kumar Sivasankar & Sriram Sridhar, Gothenburg, June 2021

NOMENCLATURE

CFD	-	Computational Fluid Dynamics
OpenFOAM	-	Open source Field Operation and Manipulation
Fe_20_3	-	Hematite
Fe_30_4	-	Magnetite
CFL	-	Courant Number
$maxCo$	-	Maximum Courant number
LKAB	-	Luossavaara-Kiirunavaara AB
LPT	-	Lagrangian Particle Tracking
FVM	-	Finite Volume Method
Ea	-	Activation Energy
PDE	-	Partial Differential Equation

SYMBOLS

τ_{ij}	-	Stress tensor
ρ	-	Density
μ	-	Dynamic viscosity
ν	-	Kinematic viscosity
S_{ij}	-	Strain rate tensor
ν_t	-	Turbulent viscosity
u_τ	-	Wall friction velocity
τ_w	-	Wall shear stress
Re	-	Reynolds number
Sh	-	Sherwood number
Sc	-	Schmidt number
Pr	-	Prandlt number
D_{12}	-	Diffusion Coefficient
γ	-	Tortuosity
ϵ_p	-	Porosity
F_c	-	Fractional conversion
d_p	-	Particle diameter
D_{eff}	-	Effective diffusion coefficient
A_{eff}	-	Model parameter
M_p	-	Molecular weight
C_p	-	Specific heat
E	-	Effectiveness factor
σ	-	Surface area factor
$\dot{Q}$	-	Heat transfer rate
h	-	Heat transfer coefficient
D	-	Darcy coefficient
F	-	forchheimer coefficient
Δ_t	-	Time scale

Contents

List of Figures

List of Tables

1
Introduction

1.1 Background

Considering environmental impacts is a primary requirement for the mining industry and for manufacturers. Mining industries have been regularly using fossil fuels and other natural resources for carrying out their respective processes. However, the sustained growth in technological advancements and alternatives to prevent excessive use of such non-renewable resources for industrial purposes is on the rise. All these improvements have led industries to pick alternative fuels in combination for the oxidation process.

One of the primary goals of LKAB (Luossavaara-Kiirunavaara AB) is to go fossil-free. To accomplish this, LKAB aims to conduct experimental analysis to evaluate the outcomes. Such experimental setups need a foundation of Computational Fluid Dynamic (CFD) models to gather ideas on its aggregates.

The production process of pellets in general at LKAB is explained as follows. Extraction of iron from its ores takes places in two steps, first is the underground mining [1], the next is the processing [2] of iron ore.

Underground mining is performed using several processes as detailed below.

- Drifting - Drifting is the process where tunnels are built to develop transport routes to mine underground ores.
- Production drilling - For the blasting process to occur, holes are drilled in the upward direction onto the ore bodies. After this, the blasting agents are charged into the holes that are drilled.
- Production loading - The removal of iron ores from the drifts is carried out by the process of loading once the blasting has been done.
- Chute transport - Chute transport is the transport of ores from rock bins to the crushers for the further crushing process to occur.
- Crushing - Once the ores arrive at the crushers, the crushers break the ores into several small pieces.
- Hoisting - Once the crushing is done, the pieces are automatically transported to the hoists, which in turn brings it to the plants for processing of the ores.

The processing of iron ores begins once the mining is performed and brought to the plants. Similar to the process of mining, the processing of iron ore is performed via

three stages which are explained as below.

- Sorting - The ores that are transferred to the sorting plant are further crushed into smaller sizes (less than 0.1 meters). Rocks in the ore are termed as residual, and they are sorted in a plant. Once a rough sorting is done, the primary sorting is performed with the help of magnetic separators as the magnetite ore is magnetic in nature. Further screening and crushing are performed to obtain the exact fractions for concentration purposes.

- Concentration is done in three steps namely, grinding and separation, flotation and, additives and filtration.

  Grinding in the concentration plant is done to get finely isolated species free from impure substances such as silicon, sodium, potassium, etc that could affect the processing. Grinding is done in multiple stages in consideration of the ore for concentration. Separation and grinding are done alternately to make sure that unwanted substances are removed from the species and desired concentrations are obtained.

  Purification of the ore is done with the aid of chemical techniques, which are termed flotation. Further, a reagent is used that binds the apatite and removes minute airlocks/bubbles from the slurry as foam.

  Additive agents such as dolomite, olivine, quartzite and limestone are included for obtaining the desired type of pellet. Further, filtration is done to get a fixed moisture fraction.

- Pelletizing - The final part of the processing of iron ore is pelletizing which constitutes three stages.

  Drying and pre-heating are the basic steps in pelletizing. It is done for two main reasons. The first being to reinforce the strength of the pellet. Secondly, it is to remove moisture.

  Sintering is performed following the drying and pre-heating process. Obtaining final properties that are desired at LKAB is done by the process of sintering. At this stage, the pellets are heated, and ore particles begin to partially melt.

  The final part is the cooling, where the pellets are rolled into balls and cooled to certain suitable temperatures and ready for steel manufacturing.

For the sake of analysing experimentally, the process of oxidizing pellets at LKAB is done in a pot furnace after which, the conversion from magnetite (Fe_3O_4) to hematite (Fe_2O_3) is analyzed. The experiment is carried out for a total duration of 3000 seconds, the initial 1610 seconds is performed to dry the pellets as they have moisture in them, the remaining 1390 seconds of the experiment is with higher oxygen being supplied with higher temperature gas to start the reaction and this process

of 1390 seconds is only considered for simulation. The data obtained from the experiment is used for validation against the numerical model developed in OpenFOAM using a Lagrangian solver, namely *reactingHeterogenousParcelFOAM*.

The selected solver is proved to be working for the moving bed of the pellet system (Which represent particles moving on a conveyor belt) which was concluded based on inputs from LKAB. Initially, this solver is configured with suitable settings to simulate a stationary bed of pellets and later verified with a moving bed of pellets.

1.2 Purpose

The simulations performed in OpenFOAMv2012 will serve as the primary data based on which we can cut time and cost factors involved in making pot furnace testing in the future with alternate fuels. The numerical model and its results will yield an idea on the performance and necessary factors that could influence the system before the commencement of further experiments. The experimental results can be anticipated based on the simulations performed on the numerical model of the pot furnace.

1.3 Limitations

The biggest limitation is how the numerical model deviates from the real-life scenario owing to the many simplifications involved in building it. The placing of pellets within the furnace in the simulation model will have a distribution mathematically defined to identify positions for performing simulations. In contrast with this, the positioning in the experimental setup will be a random phenomenon creating differences while validating results. As we deal with the Lagrangian particle tracking solver, it is necessary to generate a mesh with sizes larger than the particle size which restricts us from further refining the mesh which would have captured the flow field much more accurately.

2

Theory

2.1 Governing equations

The governing equations [3] of fluid flow derived using a control volume approach and principles of conservation of mass, momentum and energy is given below in an incompressible form with a constant viscosity assumption. The continuity equation is applied to express conservation of mass at every point in the fluid field domain. The continuity equation is given by

$$\frac{\partial v_i}{\partial x_i} = 0, \tag{2.1}$$

where v_i is flow velocity vector field and x_i indicates spatial coordinates.

The momentum equation is given by

$$\frac{\partial v_i}{\partial t} + \frac{\partial v_i v_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 v_i}{\partial x_j \partial x_j} + f_i, \tag{2.2}$$

where p denotes the pressure and t is the time coordinate. The first term on the right-hand side is the pressure gradient term, the second term is the viscous diffusion term and the third term is the body force term. Viscosity ν in the second term on the right-hand side is given by

$$\nu = \frac{\mu}{\rho}, \tag{2.3}$$

where μ is the dynamic viscosity and ρ is the density of the fluid.

The energy equation is given by

$$\rho \frac{du}{dt} = -p \frac{\partial v_i}{\partial x_i} + \Phi - \frac{\partial q_i}{\partial x_i}, \tag{2.4}$$

where u is the internal energy and q_i is given as

$$q_i = -k \frac{\partial T}{\partial x_i}. \tag{2.5}$$

Φ is expanded as

$$\Phi = 2\mu S_{ij} S_{ij} - \frac{2}{3} \mu S_{kk} S_{ii}. \tag{2.6}$$

Equation 2.5 denotes Fourier law and equation 2.6 illustrates irreversible viscous heating.

Practical flows around us are generally turbulent in nature. When the flow is turbulent, the instantaneous variables can be decomposed into their fluctuating (v',p') and its mean component of flow ($\overline{v}$,$\overline{p}$). The decomposition of velocity and pressure is given by

$$v = v' + \overline{v}, \tag{2.7}$$

$$p = p' + \overline{p}. \tag{2.8}$$

On applying the decomposition given in equation 2.7 into the continuity equation 2.1 and the decomposition of velocity and the pressure in equation 2.8 into the momentum equation in 2.2 and time averaging both the equations results in the time-averaged continuity and time averaged momentum equations. The time averaged continuity equation is given by

$$\frac{\partial \overline{v_i}}{\partial x_i} = 0. \tag{2.9}$$

The time-averaged momentum equation is given by

$$\frac{\partial \overline{v}_i}{\partial t} + \frac{\partial \overline{v}_i \overline{v}_j}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \nu \frac{\partial^2 \overline{v}_i}{\partial x_j \partial x_j} - \frac{\partial \overline{v_i' v_j'}}{\partial x_j}, \tag{2.10}$$

where $\overline{v_i' v_j'}$ is the reynolds stress tensor.

2.2 Turbulence models

The need for the turbulence model is because the flow is in the turbulent regime [4] in the cases discussed in the further sections, and it is determined based on the Reynolds number. Reynolds number is the ratio of inertial forces to viscous forces, given by $\mathrm{Re} = \frac{\rho U D}{\mu}$, where ρ is the fluid density, U is the flow velocity and D is the hydraulic diameter. Three turbulence models are investigated on simple geometries (solver verification) to check if the turbulence models affect the conversion process of Fe_3O_4 to Fe_2O_3. The turbulence models used in this thesis are limited to standard k-ϵ model, realizable k-ϵ model and k-ω SST model.

2.2.1 $k - \epsilon$ model

The k-ϵ model [5] is a widely used turbulence model. It is numerically stable and a simple model. It is a two-equation model which provides a general description of turbulence by means of two transport equations. The equation for k (turbulent kinetic energy) is given by

$$\frac{\partial k}{\partial t} + \frac{\partial (k u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\frac{\nu_t}{\sigma_k}\frac{\partial k}{\partial x_j}\right] + 2\nu_t S_{ij} S_{ij} - \epsilon. \tag{2.11}$$

The equation for ϵ (rate of dissipation of turbulent kinetic energy) is given by

$$\frac{\partial \epsilon}{\partial t} + \frac{\partial (\epsilon u_i)}{\partial x_i} = \frac{\partial}{\partial x_j}\left[\frac{\nu_t}{\sigma_\epsilon}\frac{\partial \epsilon}{\partial x_j}\right] + C_{1\epsilon}\frac{\epsilon}{k}2\nu_t S_{ij}S_{ij} - C_{2\epsilon}\frac{\epsilon^2}{k}. \tag{2.12}$$

The transport equations are given in equations 2.11 and 2.12. S_{ij} represents the rate of deformation, μ_t is the eddy viscosity. Here, the constants and its values are $C_\mu = 0.09$, $\sigma_k = 1.00$, $\sigma\ _\epsilon$=1.30, $C_{1\epsilon}$=1.44, $C_{2\varepsilon}$ =1.92.

2.2.2 Realizable $k-\epsilon$ model

Realizable k-ϵ [6] is derived from the standard k-ϵ model. This model is more viable for prediction and capturing of mean flow around complex structures which also makes it suitable for rotating flows and flow involving boundary layers with adverse pressure gradient effects. The realizable k-ϵ model equations are given by

$$\frac{\partial k}{\partial t} + \frac{\partial}{\partial x_j}(ku_j) = \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right] + P_k + P_b - \epsilon - Y_M + S_k, \tag{2.13}$$

$$\frac{\partial \epsilon}{\partial t} + \frac{\partial}{\partial x_j}(\epsilon u_j) = \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\nu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_j}\right] + \rho\, C_1 S\epsilon - C_2\frac{\epsilon^2}{k+\sqrt{\nu\epsilon}} + C_{1\epsilon}\frac{\epsilon}{k}C_{3\epsilon}P_b + S_\epsilon, \tag{2.14}$$

where the constants and its values are $\mathrm{C}_{1\epsilon} = 1.44, C_2 = 1.9,\ \sigma_k = 1.0,\ \sigma_\epsilon = 1.2$.

2.2.3 $k-\omega$ SST model

The k-ω SST model [7], a two-equation eddy viscosity model, where, SST refers to shear stress transportation. k-ω SST combines the use of the k-ω for the inner region of boundary layer and switches to the k-ϵ in the free stream region. The k and ω used in the model are the turbulent kinetic energy and dissipation rate per unit turbulent kinetic energy respectively, where, $\omega \propto \epsilon/k$.

The transport equations related to the model are given as

$$\frac{\partial k}{\partial t} + \frac{\partial (u_j k)}{\partial x_j} = P - \beta^* \omega k + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_k \nu_t)\frac{\partial k}{\partial x_j}\right], \tag{2.15}$$

$$\frac{\partial \omega}{\partial t} + \frac{\partial (u_j \omega)}{\partial x_j} = \frac{\gamma}{\nu_t}P - \beta\omega^2 + \frac{\partial}{\partial x_j}\left[(\nu + \sigma_\omega \nu_t)\frac{\partial \omega}{\partial x_j}\right] + 2(1-F_1)\frac{\sigma_{\omega 2}}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, \tag{2.16}$$

$P = \min\left(\tau_{ij}\frac{\partial U_i}{\partial x_j}, 10\beta^* k\omega\right)$,
$F_1 = \tanh\left\{\left\{\min\left[\max\left(\frac{\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{4\sigma_{\omega 2}k}{CD_{k\omega}y^2}\right]\right\}^4\right\}$,
$CD_{k\omega} = \max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i}, 10^{-10}\right)$,

where $\beta = \frac{3}{40}, \beta^* = \frac{9}{100}, \sigma_{k1} = 0.85, \sigma_{k2} = 1, \sigma_{\omega 1} = 0.5, \sigma_{\omega 2} = 0.856$.

2.3 Modified unreacted core shrinking model

The conversion process of magnetite (Fe_3O_4) to hematite (Fe_2O_3) by oxidation is a heterogeneous reaction since two phases (gas and solid) are present. To model these type of reactions, the Unreacted Core Shrinking (UCS) model was previously used as obtained from literature [8]. The limitation of this model is that it does not present a well-defined topochemical reaction behaviour. A topochemical reaction is a reaction that takes place at the boundary of solid phases. To overcome this limitation, Modified Unreacted Core Shrinking (MUCS) model is introduced [8]. Hence, our study on the reaction of the pellet bed is done using MUCS model. This model also incorporates the modelling of the mass transfer happening in the gas-solid reaction.

The chemical reaction that happens within a pellet bed is given as

$$2Fe_3O_4 + 1/2O_2 = 3Fe_2O_3. \tag{2.17}$$

The above reaction is an oxidation process for the conversion of magnetite in pellets into hematite. The MUCS model equation is given by

$$\frac{d_p}{6\alpha}F_c + \frac{d_p^2}{24D_{eff}}\left[1 + 2\left(1 - F_c\right) - 3\left(1 - F_c\right)^{2/3}\right] + \frac{d_p}{2k_{TC}\sigma E}\left[1 - \left(1 - F_c\right)^{1/3}\right] = \frac{4M_pC_{pb}}{\rho_p}t, \tag{2.18}$$

where F_c denotes the fractional conversion of magnetite, t is the time in seconds for the reaction to occur, M_p is the molecular weight of magnetite, C_{pb} is the specific heat of the tracked pellet cloud, A_{eff} is the model parameter and it gives the actual fraction of the contact area between the pellets and the gas in the pellet bed, ρ_p is the density of the tracked pellet cloud, d_p is the diameter of the tracked pellet cloud. k_{TC} is defined as thermal conductivity, whereas σ and E are the surface area factor and effectiveness factors respectively, where effectiveness factor [9] is the fraction of the particle mass on their periphery. α is the mass transfer coefficient, it is given by empirical relations [8] between particle Reynolds number, Schmidt number and Sherwood number as

$$Sh = \frac{\alpha d_p}{D_{12}} = 2.0 + 0.6\mathrm{Re_p}^{1/2}\mathrm{Sc}^{1/3}, \tag{2.19}$$

where d_p is the particle diameter and D_{12} is the binary diffusion coefficient. The particle Reynolds number (Re_p) and Schmidt number (Sc) are defined as

$$\mathrm{Re_p} = \frac{d_p U \rho_g}{\mu_g}, \tag{2.20}$$

$$\mathrm{Sc} = \frac{\mu_g}{\rho_g D_{12}}. \tag{2.21}$$

Here, U is the gas-phase velocity magnitude, ρ_g is the gas-phase density and μ_g is the gas-phase dynamic viscosity.

The term D_{eff} is the effective diffusion coefficient, which is given by

$$D_{eff} = \frac{D_{12}\epsilon_p}{\gamma}, \tag{2.22}$$

where ϵ_p is the porosity of tracked pellet cloud. The diffusion coefficient is defined as a physical constant that depends on molecule size and properties of the diffusing material along with parameters such as temperature and pressure. γ is the tortuosity of the tracked pellet cloud and it is defined as convoluted pathways of fluid diffusion through porous media.

The reaction constant K is defined as,

$$K = Aexp\left(-\frac{Ea}{RT}\right), \tag{2.23}$$

where Ea is the activation energy, which is the least amount of energy needed to bring atoms or molecules to a state where they can undergo a chemical reaction. R is the universal gas constant, T is the gas temperature and A is the pre-exponential factor.

Using the MUCS model, an ordinary differential equation system is developed to track pellet clouds in the pellet bed [8] from equation 2.18 and it is given by

$$\frac{\mathrm{d}F_c}{\mathrm{dt}} = A_{eff}\frac{4M_p c_{pb}}{\rho_p}\frac{1}{\frac{d_p}{6\alpha} + \frac{d_p^2}{12D_{eff}}\left[(1-F_c)^{-13}-1\right] + \frac{d_p}{6k_{TC}\sigma E}(1-F_c)^{-23}}. \tag{2.24}$$

2.4 Energy transfer

The energy transfer [10] in the pellet bed takes place in the form of heat transfer between the pellets and the gas. The heat transfer is explained using the convection law.

The convective heat transfer equation is given by

$$\dot{Q} = hA(\Delta T), \tag{2.25}$$

where $\dot{Q}$ is the rate of heat transfer between the particle and the fluid, h is the convective heat transfer coefficient, A and T are the surface area of the particle and

the temperature at the instant respectively.

The nusselt number relation is used to find the convective heat transfer coefficient h which is determined from the equation

$$\mathrm{Nu} = \frac{hL}{k_{TC}}, \tag{2.26}$$

where k_{TC} is the thermal conductivity of the gas phase and L is the characteristic length. The connecting equation to obtain Nusselt number is obtained from the nusselt-prandlt relation

$$\mathrm{Nu} = f(\mathrm{Re}, \mathrm{Pr}). \tag{2.27}$$

Using the Nusselt-Prandlt-Reynolds relation for spherical objects in a flow [11], Nusselt number is determined using the equation

$$\mathrm{Nu} \ = 2 + 0.4\,\mathrm{Re_p}^{1/2}\,\mathrm{Pr}^{1/3}\,, \tag{2.28}$$

where, Pr is the Prandlt number, which is the ratio of momentum diffusivity to thermal diffusivity. The prandlt number is given by

$$\mathrm{Pr} = \frac{C_p \mu}{\rho}, \tag{2.29}$$

where, μ is the dynamic viscosity, ρ is the density and C_p is the specific heat.

The transfer of heat from one phase to the other is described in the form of source terms in continuum equations for energy transfer. The temperature of the particle is updated at every time step, dt in equation

$$\dot{Q} = hA(\Delta T)dt, \tag{2.30}$$

hence, to evaluate the overall $\dot{Q}$, we go for an energy balance between heat transfer and internal energy of the cloud to obtain overall heat that is transferred from the particle cloud to the fluid phase as

$$mC_p \frac{dT}{dt} = \dot{Q}_{Total}. \tag{2.31}$$

Using the equations 2.30 and 2.31, we finally obtain

$$mC_p \frac{dT}{dt} = hA(\Delta T). \tag{2.32}$$

2.5 Discretization

As we approach the system of Partial Differential Equations (PDE) using the Finite Volume Method (FVM) [12], discretization over each control volume in a domain is carried out. It is a technique used in FVM to arrive at the algebraic equations from the PDE from the physical laws of conservation.

2.5.1 Time discretization

The update of the solution respective to time is determined by the time discretization. The implicit Euler scheme is used to discretize the time derivative component $\frac{\partial \varphi}{\partial t}$.

$$\frac{\partial \varphi}{\partial t} = \frac{\varphi(t + \Delta t) - \varphi(t)}{\Delta t} \tag{2.33}$$

Equation 2.33 gives the discretization of $\frac{\partial \varphi}{\partial t}$ using the implicit Euler scheme. Implicit Euler scheme is first order accurate and bounded, meaning the value of φ is always bounded by its boundary values [12].

2.5.2 Gradient terms

The gradient terms in the equation systems are discretized using central-differencing scheme [12]. In central-differencing scheme the values at the face of control volume are estimated using linear interpolation. This scheme is second-order accurate.

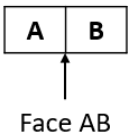


Figure 2.1: Two control volume.

Figure 2.1 shows two equidistant control volumes with cell A and B, and their shared face AB. In general for central differencing scheme, value of any transport variable Φ_{AB} at faces is given by

$$\Phi_{AB} = f_x \Phi_A + (1 - f_x)\Phi_B, \tag{2.34}$$

where Φ_A and Φ_B represent the transport variables at the cells A and B respectively. For a uniform grid f_x is 0.5, which gives

$$\Phi_{AB} = \frac{\Phi_A + \Phi_B}{2}. \tag{2.35}$$

2.5.3 Divergence terms

The divergence terms in the equation system are discretized using first order upwind scheme [12]. In this scheme the value of any variable at the face of the cell depends on the flow direction. This scheme is very effective for strong convective flows. In general for the upwind scheme, value of Φ at faces is given by

$$\Phi_{AB} = \Phi_A, if F > 0, \tag{2.36}$$

$$\Phi_{AB} = \Phi_B, if F < 0, \tag{2.37}$$

where the result is the value at a face between the nodes A and B. F is the flux, where, $F = \rho u_i \Delta S$, where S is the surface.

2.6 Courant number

Courant number measures the rate at which the information is transferred under the influence of a flux field. It also acts as a limiting factor for the performance of a numerical scheme. It can be used to evaluate the time step (deltaT) required in a transient simulation based on the mesh size. Courant number is given by [13]

$$Co = \Delta_t \tau, \tag{2.38}$$

where Δ_t is the time-step size and τ is the characteristic time scale which is given by

$$\tau = \frac{1}{2V} \sum_{faces} |\phi_i|, \tag{2.39}$$

where V is the volume of the cell, ϕ is face volumetric flux and the summation is done over all the cell faces.

2.7 Lagrangian particle

The pellets in the pot furnace and in the cases used for solver verification are treated as lagrangian particles [14]. The pellets are treated as such because individual particle information is necessary to compute mass fractions of the pellets after the conversion process. The concept from lagrangian particle tracking of cell sizes being larger than the particle sizes is used to position the bed of pellets within the domain. However, no tracking of pellets is done as they remain stationary in the pot furnace. One-way coupling is used, where the effect of pellets on the flow field is neglected.

2.8 Porous zone

The porous zone [15] can be added to a fluid domain using $Darcy - Forchheimer$ equation without any additional cost. The Darcy–Forchheimer equation is an empirical formula that links the pressure loss due to friction across a porous medium to the velocity of the flow within the medium. The equation acts as a sink term in the Navier-Stokes equation as given by

$$\frac{\partial(\gamma_z v_i)}{\partial t} + \frac{\partial v_i v_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \nu \frac{\partial^2 v_i}{\partial x_j \partial x_j} + S_i, \tag{2.40}$$

where γ_z is the porosity of the zone and its value is between 0 and 1. For a value of 0, there is no porosity effect while it becomes completely porous when it is 1. The last term on the right-hand side denotes the source term. The source term acts as a sink due to the presence of the negative sign.

$$S_i = -\left(\mu D_{ij} + \frac{1}{2}\rho \left|v_{kk}\right| F_{ij}\right) v_i, \tag{2.41}$$

This is the $Darcy - Forchheimer$ equation given in equation 2.41, with D_{ij} and F_{ij} as Darcy and Forchheimer coefficients respectively. For a homogeneous porous zone,

the coefficients are expressed as scalars and the source term for the homogeneous porous zone is given below

$$S_i = -\left(\mu D + \frac{1}{2}\rho \left|v_{jj}\right| F\right) v_i. \tag{2.42}$$

2.8.1 Solution algorithm - PIMPLE

Solution algorithm is a set of procedures to solve a problem, in case of CFD it the discretised equation system. The PIMPLE algorithm [16] is used for the verification cases and pot furnace simulations as well. It is a combination of the PISO (Pressure Implicit with Splitting of Operators) and SIMPLE (Semi-Implicit Method for Pressure Linked Equations) algorithms. The PIMPLE is used for the purpose of transient simulations. The understanding of the PIMPLE algorithm is that the SIMPLE algorithm is implemented at every step, with outer *nOuterCorrectors* being the iterator, where *nCorrectors* is the number of times the pressure correction is happening. As we attain convergence, we shift to the next time step and keep working until a completed solution is obtained.

3
Methodology

This section will describe the software used and the case setup specific to this master thesis. The verification process of the solver to work on a stationary bed of pellets is also mentioned. Further, the meshing process of the pot furnace is discussed.

3.1 OpenFOAM

OpenFOAM is a C++-based open-source CFD software package developed by OpenCFD Ltd. OpenFOAM is the abbreviation for "Open-source Field Operation And Manipulation". The wide range of solvers helps to solve continuum mechanics problems, especially CFD problems. Also, the utilities within OpenFOAM give a variety of pre-processing and post-processing functions to manipulate the data from the simulation. Due to the absence of a graphical user interface, the user must be aware of the file structure inside OpenFOAM. The files within the case setup vary depending on the chosen solver as different solvers require different inputs. Further in this report, the file structure of the case setup required by *reactingHeterogenousParcelFoam* will be specified.

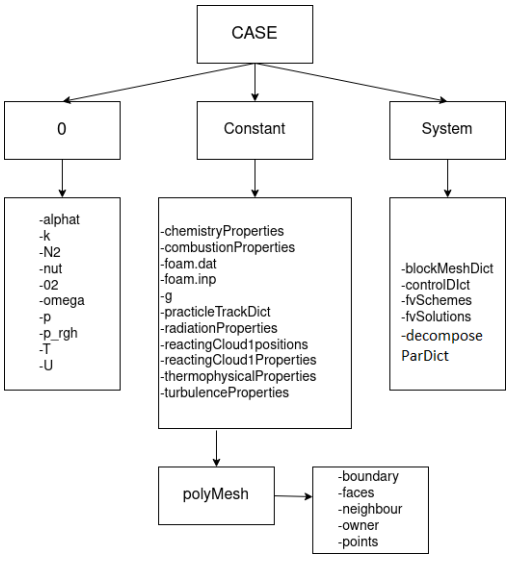


Figure 3.1: File structure.

3.1.1 Time directory

The time directory has data for each fields in individual files [17]. The data in those files can either be initial values and boundary conditions specified by the user in 0 directory or written by OpenFOAM in other time directories when solution is computed. The directories are named after each time step and the write interval can be specified by the user.

3.1.2 Physical properties directory - Constant

This directory consists of the physical setup specific to the application. In particular to the *reactingHeterogenousParcelFoam* solver, the following physical properties are specified.

- File *chemistry properties*, *combustion properties*, and *radiation properties* specify the respective models to be used and values specific to the model selected will be mentioned. Also they can be turned ON or OFF.
- File *g* specifies the acceleration due to gravity.
- File *turbulence properties* specifies the turbulence model that will be used.
- File *practicleTrackDict*, contains the properties of the particles that are to be updated.
- File *thermophysical properties* consists of the inert species, liquid, gas, and solid species present in the system. Additionally the chemical reaction underwent by the elements is mentioned in this file.
- File *reactingCloud1Positions* has the initial position of the particles present in the domain. This file is specific to the manual injection model which will be discussed in further sections.
- File *reactingCloud1Properties* consists of all the properties and models required for the lagrangian cloud of particles, the particle forces, injection model, collisional model etc.
- File *fvOptions* contains equation systems available in OpenFOAM that can be run-time manipulated. In this project, Darcy–Forchheimer equation is used.

3.1.2.1 Mesh description directory - polyMesh

After the meshing is executed, a new folder called *polyMesh* will be generated inside the *constant* folder. This folder contains the description of the mesh.

- File *points* has the list of all the points in the mesh.
- File *faces* contains the list of faces with the points which describe each face.
- File *neighbors* has information on cell connections.
- File *owners* gives information on faces owned by a cell.
- File *boundaries* gives information about specific boundaries such as the inlet, outlet and walls for each patch created while meshing.

3.1.3 Simulation settings directory - System

The system directory contains the simulation control parameters. Here, the parameters for the solution procedure can be set. The following is the description for each

file inside the system directory.

- File *blockMeshDict* contains the details of the blocks created to generate a mesh. The details include the vertices, edges, blocks including the number of cells and grading level. Also, this file is where the type of boundary is defined and is assigned to the patches. Alternate meshing methods such as the *SnappyHexMesh* or *cfMesh* can be used and its corresponding mesh dictionary replaces the *blockMeshDict*.
- File *decomposeParDict* describes the decomposition method and the user determines the number of parts the mesh is to be decomposed into for running a parallel simulation.
- File *controlDict* has information on the control properties for the simulation such as the time step, end time, output, write interval etc. The courant number can be limited using *maxCo* option combined along with the *adjustTimeStep* option turned ON.
- File *fvSchemes* is used to specify the numerical schemes for the derivatives in the equations such as the time derivative, divergence, gradient etc.
- File *fvSolution* is used to specify the type of internal solver with its tolerances for the fields like pressure, velocity etc. Also, the algorithms such as the PIMPLE, SIMPLE, and PISO are assigned with their respective under relaxation factor.

3.2 Solver - reactingHeterogeneousParcelFoam

The solver used is the *reactingHeterogenousParcelFoam* [18]. The *reactingHeterogenousParcelFoam* is a transient solver for compressible and turbulent flow situations with reacting multiphase particle clouds. This solver has been chosen specifically as we deal with a multiphase and transient solver that incorporates multiphase lagrangian directory, which includes the properties of the particle cloud. The *reactingHeterogeneousParcelFoam* source file is given below.

```
/*---------------------------------------------------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     |
    \\  /    A nd           | www.openfoam.com
     \\/     M anipulation  |
-------------------------------------------------------------------------------
    Copyright (C) 2018-2019 OpenCFD Ltd.
-------------------------------------------------------------------------------
License
    This file is part of OpenFOAM.
Application
    reactingHeterogenousParcelFoam
Group
    grpLagrangianSolvers
```

```
\*---------------------------------------------------------------------------*/
#define CLOUD_BASE_TYPE HeterogeneousReacting
#define CLOUD_BASE_TYPE_NAME "HeterogeneousReacting"

#include "reactingParcelFoam.C"

// ************************************************************************
```

Listing 3.1: reactingHeterogeneousParcelFoam.C.

As we observe in the listing 3.1 for reactingHeterogenousParcelFoam.C, it includes the original source file of the solver reactingParcelFoam.C. Also, the cloud type and name are mentioned as HeterogeneousReacting, so that the solver used is reactingParcelFoam with a heterogeneous reacting cloud. The cloud contains the reacting particles specifically pellets in our case.

3.3 Injection model - Manual injection

An injection model is used to introduce a particle or a group of particles into the computational domain. After a thorough investigation of the available injection models, manual injection model is chosen with the *setPositionAndCell* method. The inputs are given in *constant/reactingCloud1Properties* in *subModels/injectionModels* as follows.

- The injection type is given under this section and in our case it is *manualInjection*.
- The total mass injected in the system is mentioned in *massTotal*. In the cases described in the further section, the total mass is calculated based on the density, geometry and the number of particles injected in the system.
- In *parcelBasisType*, the available options are mass, number, and fixed type. In the cases described in the further section, fixed type is used and the number of particles is given in *nParticle* with the value set to 1, as we simulate individual particles.
- Input *SOI* is the start time of injection and is set to 0 as all the particles are present from the beginning of the simulation.
- The size of particles is mentioned in *sizeDistribution* and the *fixedValuedistribution* option is selected with a particle diameter of 0.011 m.
- The position of the particles is mentioned in a file called *reactingCloud1Positions*.

The above mentioned information within the directories are specific to the kind of model, solver and algorithm selected. Further information on available options and alternatives can be found in the OpenFOAM user guide [19]

3.4 Solver verification

In this part, the solver verification is performed by comparing a fixed bed of pellets (pellets are the particles in our case) system with a moving bed of pellets system. Based on the input from LKAB, it is found that the solver *reactingHeterogenousParcelFoam* is able to capture the conversion process of magnetite to hematite for a moving bed of pellet system. Pot furnace involves a fixed bed of pellets. Hence, a verification study is done by using the solver on a fixed bed of pellets and compared with the moving bed of pellets system in a smaller domain. The conversion process between the two systems is analyzed and also verified with different turbulence models to conclude the working of the solver with a fixed bed of pellets and further put to use for the pot furnace simulations.

3.4.1 Meshing and boundary conditions

The dimensions of the geometries studied for the verification process is given below. The meshing is performed in OpenFOAM-v2012 using the *BlockMesh* feature.

Table 3.1: Geometry dimensions and cell count

	Dimension			**Cell count**			
	x(m)	**y(m)**	**z(m)**	**x**	**y**	**z**	**Number of pellets**
Fixed bed	0.1	0.1	0.01	9	9	1	8
Moving bed	0.9	0.1	0.01	75	9	1	8

Table 3.1 gives the dimension of the geometry and the cell counts in each direction. A thickness of 0.01m with 1 cell across the z-direction is specified in both the cases as OpenFOAM uses a 3 dimensional mesh for 2 dimensional cases too.

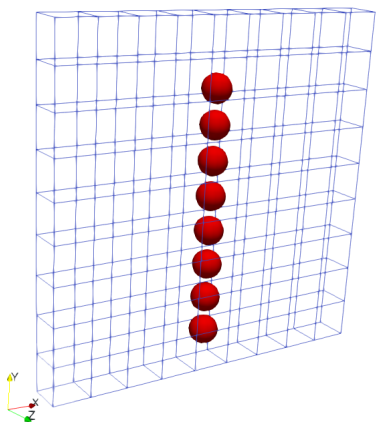


Figure 3.2: Mesh for the fixed bed simulation with pellets.

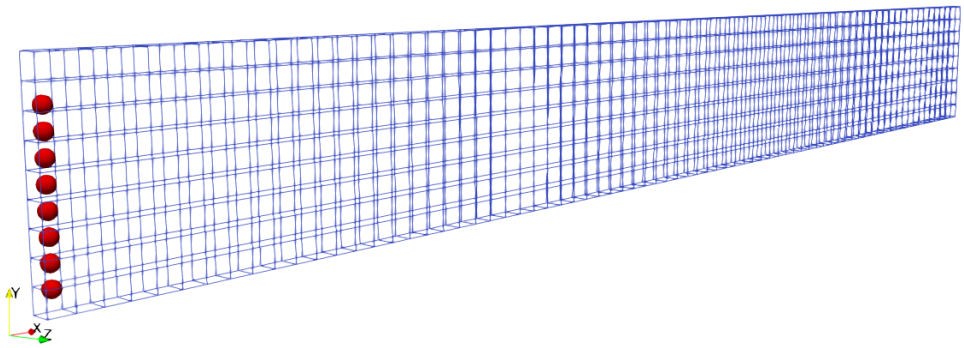


Figure 3.3: Mesh for moving bed simulation with pellets.

The boundary conditions that are used for the verification cases, which were derived from the previous moving bed of pellet system from LKAB are given below.

Table 3.2: Boundary conditions

Boundary	u (m/s)	T (K)	O2 (%)
Inlet	Velocity inlet (0, -5.5, 0)	uniformFixedValue 600	uniformFixedValue 20
Outlet	PressureInletOutletVelocity	inletOutlet	inletOutlet
Wall	Fixed value (0, 0, 0)	zeroGradient	zeroGradient

Table 3.2 gives the boundary conditions for the inlet, outlet and the wall boundaries for the velocity, temperature and oxygen volume percentage fields. A constant velocity of 0.003m/s is initialized to the pellets in the moving bed of pellet system.

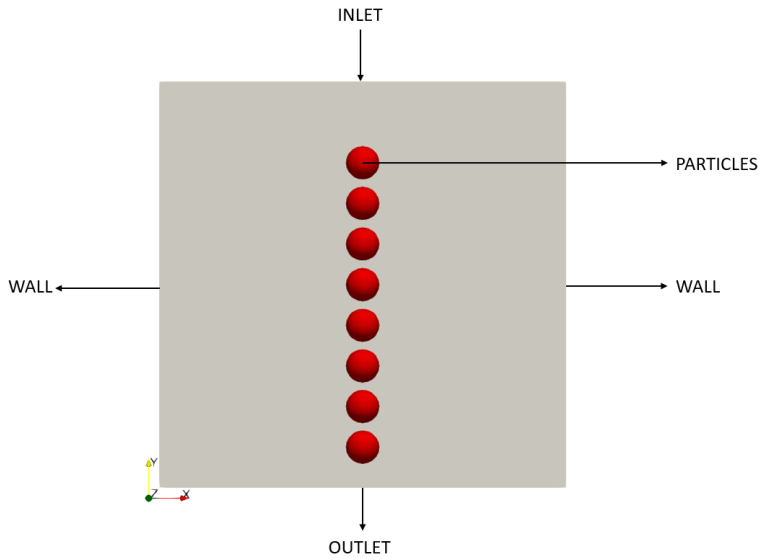


Figure 3.4: Boundary conditions for fixed bed simulation.

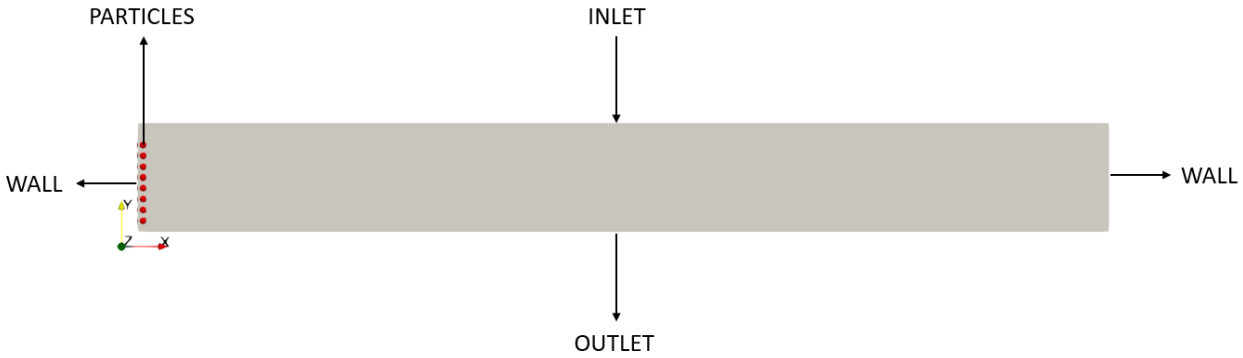


Figure 3.5: Boundary condition for moving bed simulation.

3.4.2 Physical properties

As we introduce pellets into the domain, it is necessary to define the physical properties of the pellets. The thermophysical and reaction models in OpenFOAM requires some input data, which will be given in this section.

Table 3.3: Thermophysical properties of magnetite and hematite

Property	Notation	Fe_3O_4	Fe_2O_3
Density (kg/m^3)	ρ	5100	5100
Specific heat (J/kgK)	C_p	850	710
Thermal conductivity (W/mK)	k_{TC}	0.04	0.04
Heat of reaction (J/kg)	H_f	0	5.25e5
Molecular weight (kg/kmole)	W	232	159

In *thermophysicalProperties*, N2 is mentioned as inert species. The two solids are Fe_3O_4 and Fe_2O_3 and their respective properties are specified in table 3.3 obtained from [20].

Table 3.4: MUCS coefficients

MUCS Coefficients	**Values**
$D_{12}(m^2/s)$	2.724e-4
ϵ_p	0.41
γ	3.07
σ	1
E	1
A (m/s)	3.14e4
A_{eff}	0.7
Ea (J/kmol)	1.651e5

The coefficients corresponding to the MUCS model are specified in table 3.4 and taken from [20] as well. The mass fractions of the phases, gas, liquid and solid are

set to 0, 0 and 1 respectively, and the mass fractions of Fe_3O_4 and Fe_2O_3 are set to 1 and 0 respectively in *reactingCloud1Properties* since the pellets comprise of solid Fe_3O_4 at the start of the simulation. The effect of gravity is not included for the verification cases, as in the experiment the pellets are either moving on a conveyor for a moving bed or kept fixed over a plate for the fixed bed case.

3.4.3 Verification

The verification study between two simulations is done for the conversion from Fe_3O_4 to Fe_2O_3. The indicator of this conversion process is the reducing mass fraction of Fe_3O_4 and increasing mass fraction of Fe_2O_3 simultaneously within the domain. Here the mass fractions are taken to be the average mass fraction of all pellets placed.

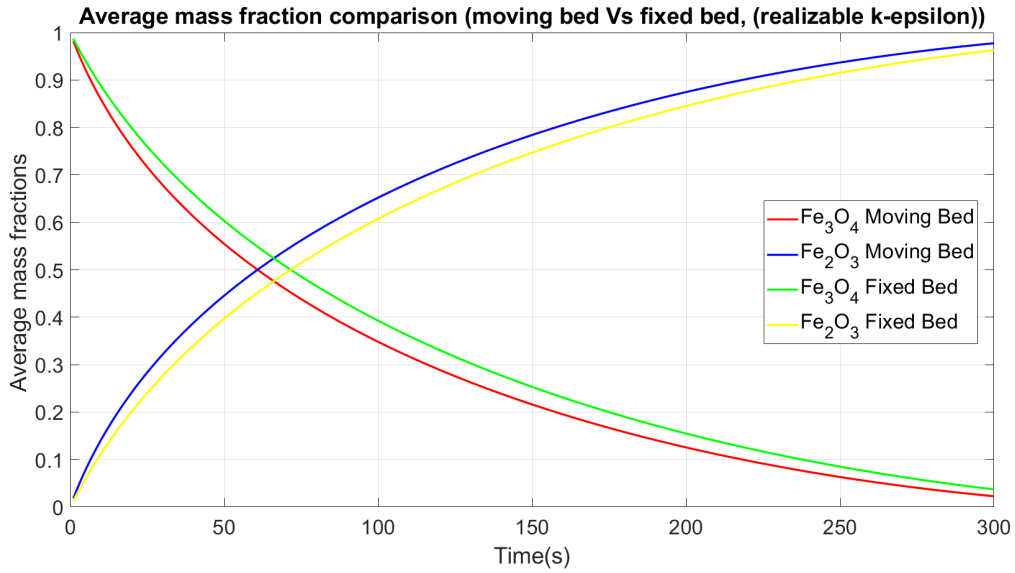


Figure 3.6: Average mass fraction for realizable k-ϵ turbulence model.

As the realizable k-ϵ model was proved to be working for the previous moving bed of pellet simulations from LKAB, this model is chosen for the initial verification between the fixed and moving bed. It can be inferred from figure 3.6 that the conversion of magnetite to hematite has completely occurred in the fixed bed simulation and the trend is similar to that of the conversion occurring in the moving bed case. Even though the trend is similar, there exists a slight difference in curves due to the fact that the pellets are fixed in the former, whereas moving in the latter case. This helps to infer that the chosen solver used works on the fixed bed of pellets.

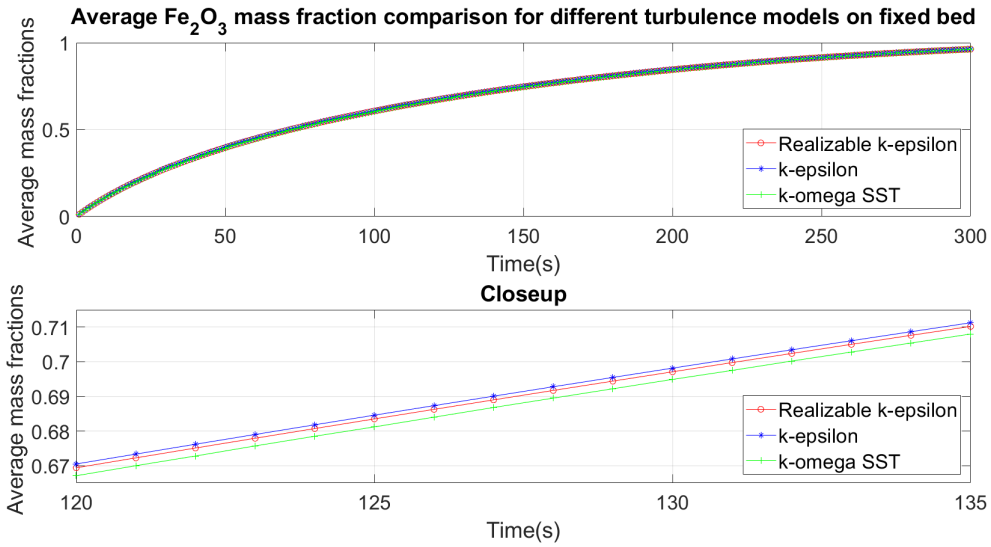


Figure 3.7: Average Fe_2O_3 mass fraction comparison for different turbulence model.

As the realizable k-ϵ works for the fixed bed too, now the zone of interest lies upon the effect of other chosen turbulence models. Hence, the fixed bed of pellet system is tested with the k-ϵ and k-ω SST models and compared with the realizable k-ϵ model in figure 3.7. From we can infer that the effect of different turbulence models is negligible on the conversion process as the relative difference between the curves are negligible. Hence, the realizable k-ϵ model that was used for the moving bed of pellets by LKAB is further used for the pot furnace simulations.

3.5 Meshing - Pot furnace

The primary task before the simulation is to a generate a mesh for the pot furnace. Hence, a mesh that suits the detailed geometry model was developed using *SnappyHexMesh*. The geometry with the mesh and the roster plate is described in the figures below.

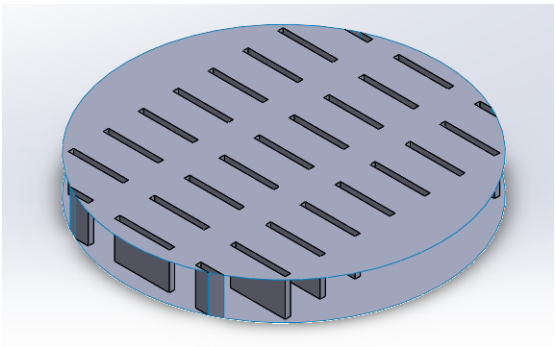


Figure 3.8: Roster plate.

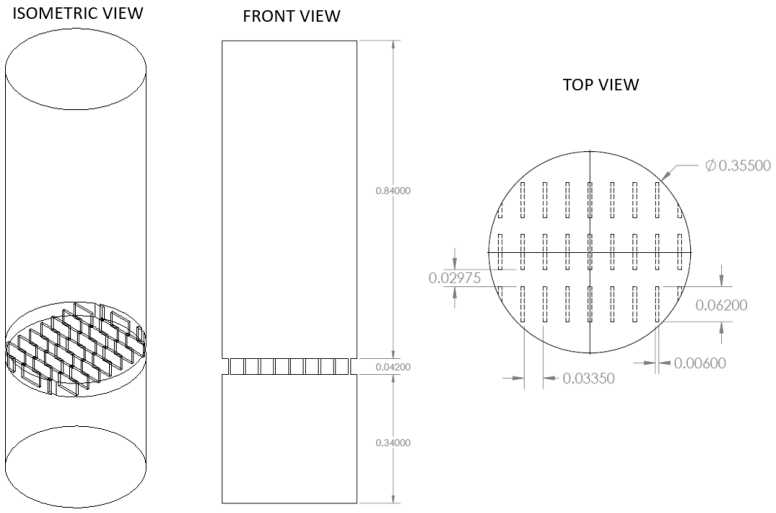


Figure 3.9: Geometrical dimensions of pot furnace.

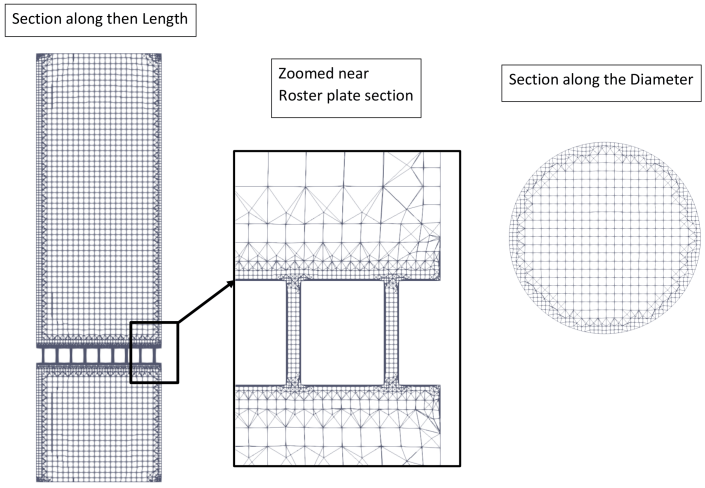


Figure 3.10: Mesh with roster plate.

Figure 3.10 gives the cross-sections of the pot furnace geometry that is meshed using the *SnappyHexMesh*. The experimental pot furnace uses a roster plate (in the figure 3.8), a metal plate with slots over which the bed of pellets is placed. It can be observed in the figure 3.9 that the size of the slots in roster plate are quite small compared to the size of the zones above and below. The complexity begins when the cell sizes are small inside the slots. Also, the cell sizes should be larger than the pellet sizes as discussed in section 2.7. This makes it difficult to create a mesh with a smooth transition between the slots and the cylinder without further refinements. Also due to a poor transition and small cell size inside the slots, the Courant number attains a high value in these regions. To account for the high Courant number, the deltaT is very low which in-turn makes the computational time longer.

3.5.1 Implementation of porous zone

To overcome the problem of a long computational time, a porous zone is implemented to model the flow through the roster plate as a replacement in the exact same zone. The mesh with the porous zone replacing the roster plate is obtained using the *BlockMesh* feature in OpenFOAM. The domain is split into three cylindrical zones - bottom, porous and top zones. The diameter of all three zones are the same while their lengths vary. Each cylindrical zone is meshed by creating five blocks using the O-grid method, by modifying and running the *M4 macro* from [21]. Finally each zone is connected with common vertices. The blocks in porous zone are defined as *porosity* in *blockMeshDict*. The porosity properties are then defined using Darcy-Forchheimer equation in *fvOptions* as follows.

```
porosity1
{

    type            explicitPorositySource;

    explicitPorositySourceCoeffs
    {
        selectionMode   cellZone;

        cellZone        porosity;

        type            DarcyForchheimer;
        d   (1.35e7 1.35e7 1.35e7);
        f   (0 0 0);

        coordinateSystem
        {
            origin  (0 0 0);
            e1      (1 0 0);
            e2      (0 1 0);
        }
    }
}
```

Listing 3.2: fvOptions.

In listings 3.2, d and f corresponds to the D and F coefficients described under section 2.8. The coefficient values are based on inputs from LKAB. A local coordinate system is required to implement a porous zone. In this case the origin of the local coordinate system and the computational domain is same. The vector e1 and e2 is set along x-axis and y-axis respectively, then the vector e3 is created automatically based on the right hand rule for coordinate systems.

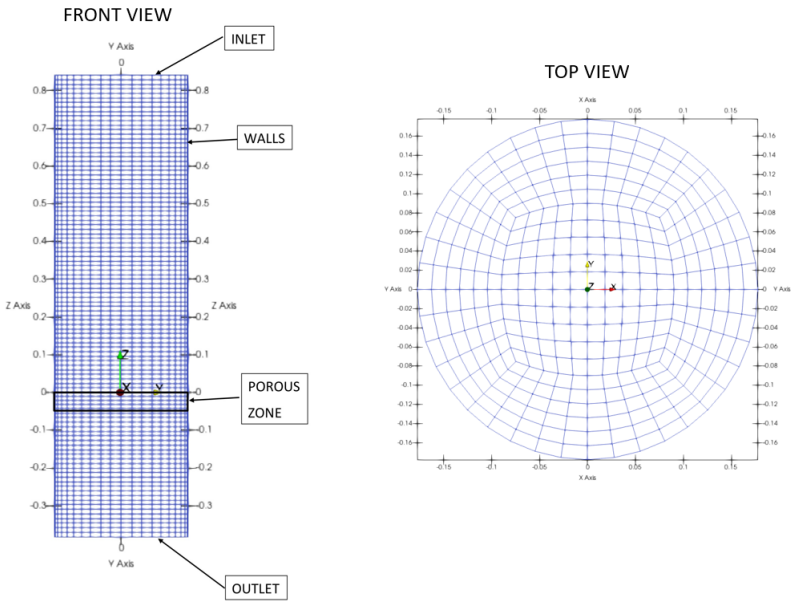


Figure 3.11: Mesh cross-sections with boundaries.

Figure 3.11 gives the mesh with a porous zone and its boundaries. The top view of the mesh is illustrated to support the O-grid method of meshing the cylinder.

3.6 Boundary conditions and physical properties

For the pot furnace simulations the boundary types are the same as mentioned in table 3.2. The values for velocity, temperature and oxygen volume percentage are time varying in the experiment. So time varying boundary conditions are implemented in U,T and O2 files in *0* directory using *uniformValue table* option available within OpenFOAM. The time varying values are extracted from the experimental data from LKAB. The following tables consists of the values of velocity, temperature and oxygen volume percentage varying with time.

Table 3.5: Time-varying boundary conditions

T	
Time (s)	K
0	473
121	473
122	773
362	1073
1390	1073

O2	
Time (s)	vol %
0	19
110	19
140	1.5
665	1.5
670	42.8
1390	42.8

U	
Time (s)	m/s
0	(0, 0, -2.965)
10	(0, 0, -2.965)
20	(0, 0, -1.89)
147	(0, 0, -1.89)
152	(0, 0, -1.3)
405	(0, 0, -2.1)
1390	(0, 0, -2.1)

The first 1610 seconds of the 3000 seconds in the experiment, air at temperature 423K is passed through the domain to remove moisture from the pellets. So the

internal field temperature is set to 423K in the simulations.

The physical properties pertaining to the pot furnace simulation are similar to solver verification as in section 3.4.2.

3.7 Pellet positioning

In the experimental pot furnace the pellets are randomly placed. A random positioning in simulations cannot be performed as we are constrained to the conception of cell size being larger than pellet size as described in section 2.7. Hence, positioning to obtain an orderly arrangement is performed. The initial step is to identify the required number of pellets, for a requirement of hundred kilograms of pellets of magnetite with a density of 5100 kg/m^3. Hence, the estimation of the required number of pellets is carried out using the relation $\rho = \frac{M}{V}$, where ρ is the density of magnetite, V is the volume (sphere of diameter 0.011m) and M is the mass of one pellet. Further the number of pellets for hundred kilograms of pellets is estimated to be 28656.

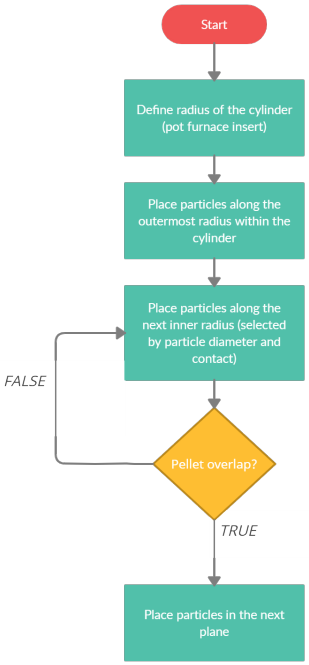


Figure 3.12: Pellet positioning algorithm.

The flowchart in figure 3.12 illustrates the algorithm that explains the idea behind the positioning of pellets within the pot furnace. Positioning is performed for one plane and repeated to a height, until the number of pellets for a hundred kilogram mass of magnetite is achieved. While placing the pellets in the outermost radius, the placing is done in such a manner that the complete pellet body is within the cylinder. The next inner radius to place pellets is chosen by the diameter of the pellet, where the pellets are in contact externally with other pellets in the outermost

radius. This pattern is repeated until we move into the innermost radius where a overlap of pellet occurs. Pellet overlap refers to the intersection of the pellet bodies which is constrained. The next plane to place pellets is also chosen similar to the selection of the inner radius, considering the diameter of pellet and external contact.

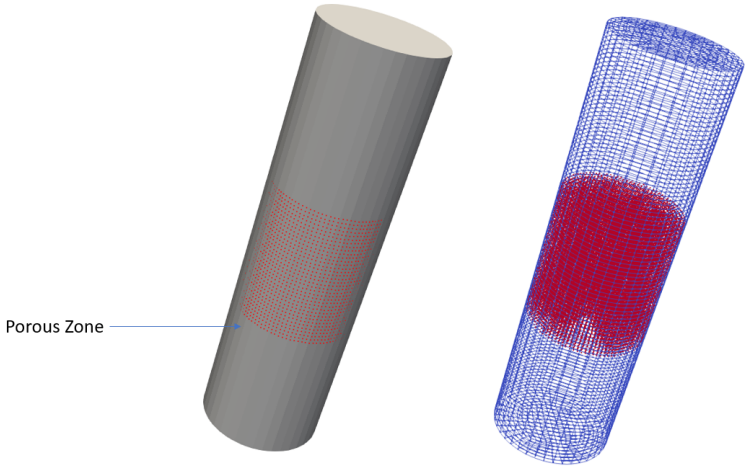


Figure 3.13: Pellets in pot furnace above porous zone.

Once the positioning of pellets is performed, the positions are fed to *reactingCloud1Positions* file. The positions fed are present above the porous zone. The positions are incorporated and shown in figure 3.13. As the roster plate contains a physical blockage of pellets from falling off due to the effect of gravity, the same effect is given to the porous zone by removing the gravitational force effect from the pellets.

3.8 Factors influencing convergence and simulation time

From section 1.1 we know that the duration of the simulation is 1390 seconds. This is a long physical time, so to speed up the simulation process there is a need for longer time steps. When *adjustTimeStep* is turned on, the CFD code dynamically adjusts the time step (deltaT) of the simulation based on the maximum Courant (CFL) number limit mentioned and this is done to prevent the simulation from crashing due to large time steps for a transient simulation. The following study on Courant number and deltaT independency is performed on pot furnace with pellets for a duration of 50 seconds of physical time to check if the physics of the simulation is captured properly even with a larger CFL and deltaT.

3.8.1 Courant number independence study

The following study is performed to determine whether a Courant number value of more than 1 captures the physics of the simulation without losing any details,

specifically mass fractions in our case. Courant number can be controlled by setting a limit, using the *maxCo* option in *system/controlDict*. The study is performed for Courant numbers 1, 5, and 10 and the mass fraction of hematite (YFe_2O_3) is plotted and compared. The relative difference percentage is calculated with respect to the simulation with maximum courant number 1 and tabulated in the table 3.6.

Table 3.6: Relative difference in % for different Courant numbers (MaxCo)

Max CFL Number	Relative difference %
1 vs 5	10.41071
1 vs 5, with 5 nCorrectors	0.542036
1 vs 10, with 5 nCorrectors	0.745774

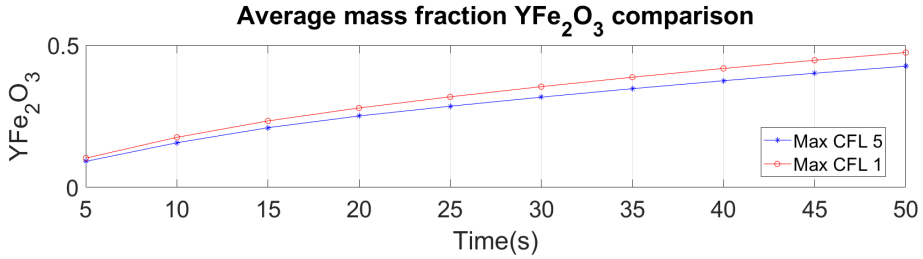


Figure 3.14: Comparison of average mass fraction of Fe_2O_3 for different max Courant number.

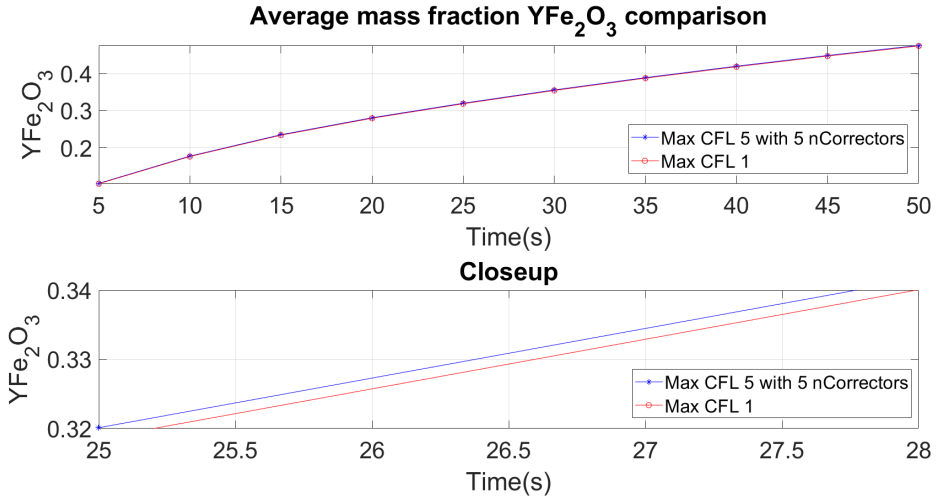


Figure 3.15: Comparison of average mass fraction of Fe_2O_3 for different max Courant number.

From figure 3.14, it is observed that there is a deviation of the mass fraction when a *maxCo* of 5 is set and the relative difference is observed to be 10.41 %. By further modifying the *nCorrectors* of the PIMPLE solver and setting it to 5, the deviation

is reduced which can be seen in the figure 3.15 along with the relative difference percentage also being reduced to 0.54%

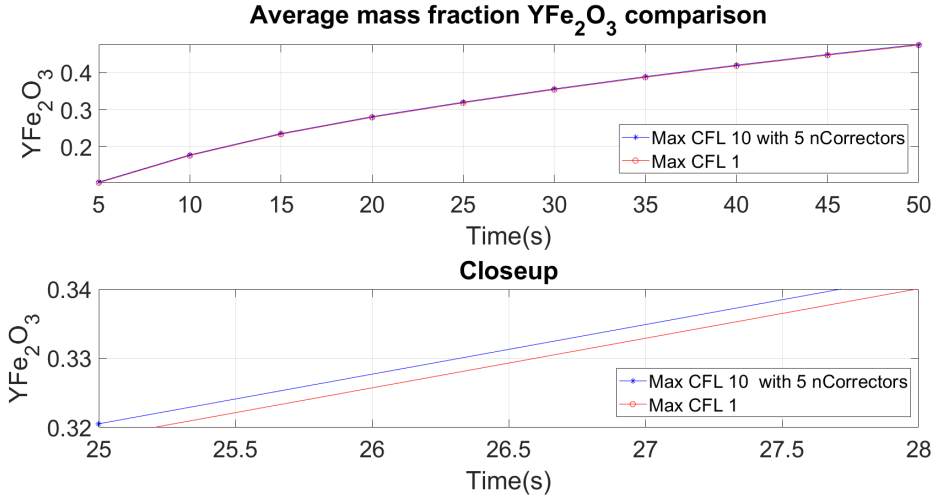


Figure 3.16: Comparison of average mass fraction of Fe_2O_3 for different max Courant number.

When the *maxCo* is set to 10 with the default number of *nCorrectors* (2) of the PIMPLE solver, the simulation diverges. After increasing the *nCorrectors* to 5 and performing the simulation, the mass fraction comparison is plotted in figure 3.16 gives an error of 0.74%

Based on this study, it is observed that the *maxCo* of 5 with 5 *nCorrectors* captures details on mass fractions closer to the *maxCo* 1 with minimal relative difference out of the investigates courant numbers. So to reduce the simulation time and also to capture proper physics, maximum Courant number of 5 is used for further simulations.

3.8.2 Time step independence study

The following study is performed to determine if the simulation results are independent of the maximum allowed time step which can be set using the *maxDeltaT* option in *system/controlDict*. The *adjustTimeStep* is used in this part of the study too.

The study is performed for the following max deltaT values 0.1, 0.01, 0.001 and 0.0001. By selecting the shortest time step (0.0001) for the base simulation, the relative difference percentage for the mass fraction of hematite (YFe_2O_3) with other time steps is compared and tabulated in table 3.7, and plotted in the following figures too.

Table 3.7: Error percentage for different Max deltaT

Max deltaT	Relative difference %
0.0001 vs 0.1	10.79041
0.0001 vs 0.1, with 5 ncorrectors	0.11485
0.0001 vs 0.01	10.82591
0.0001 vs 0.01, with 5 ncorrectors	0.308613
0.0001 vs 0.001	0.353925

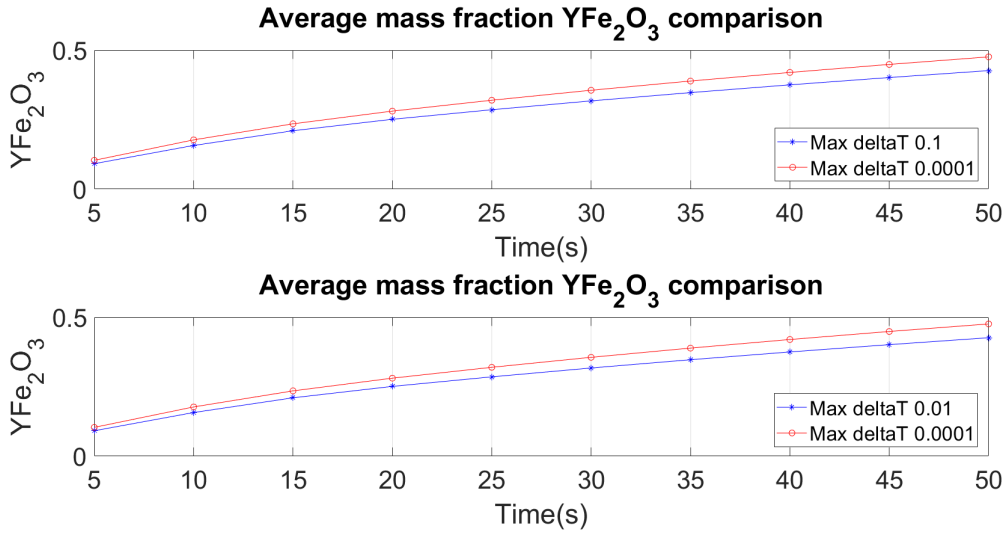


Figure 3.17: Comparison of average mass fraction of Fe_2O_3 for different max deltaT.

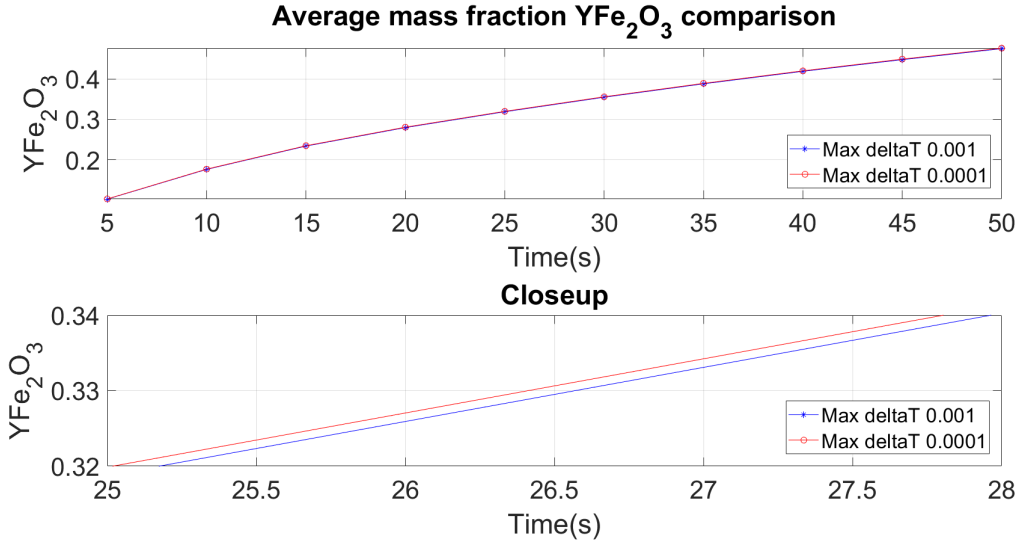


Figure 3.18: Comparison of average mass fraction of Fe_2O_3 for different max deltaT.

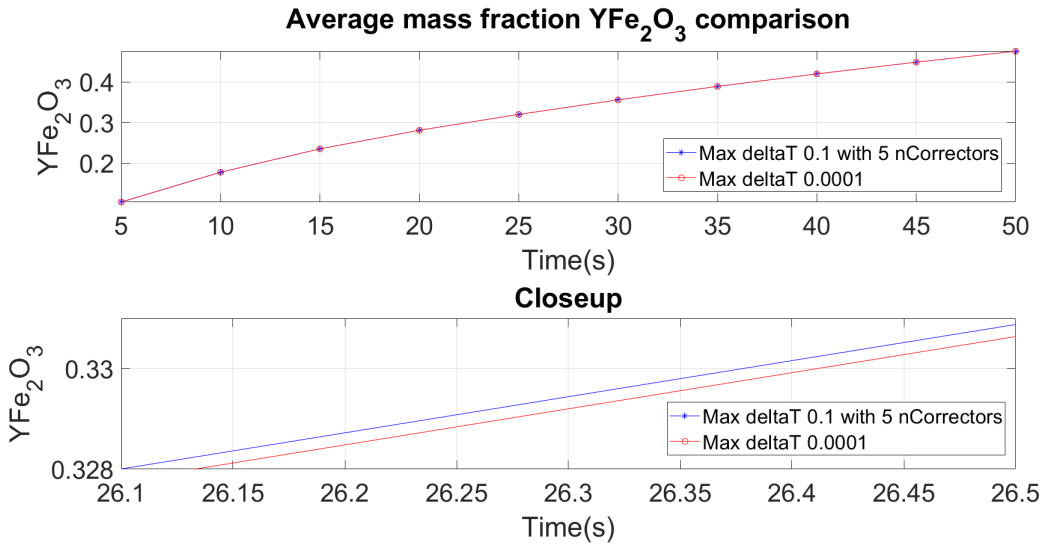


Figure 3.19: Comparison of average mass fraction of Fe_2O_3 for different max deltaT.

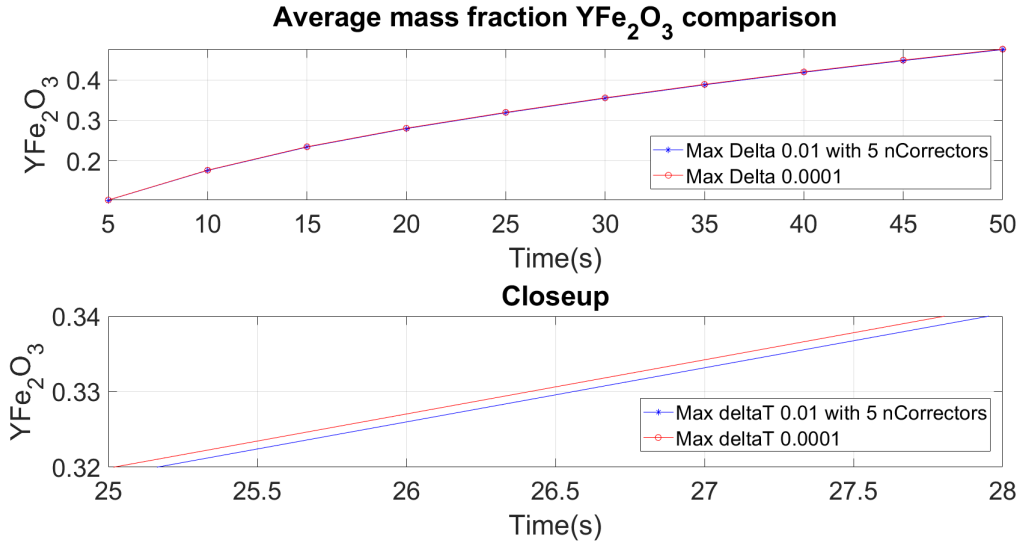


Figure 3.20: Comparison of average mass fraction of Fe_2O_3 for different max deltaT.

The figure 3.17 shows the comparison of mass fractions of Fe_2O_3 for maximum deltaT 0.1 vs 0.0001 and 0.01 vs 0.0001, from which it can be observed that the deviation in mass fraction is large and the difference is about 10.79% and 10.82% respectively.

It can be observed from the figure 3.18 that the mass fraction comparison for max deltaT 0.001 vs 0.0001 is similar and the relative difference is quite low with a value of about 0.355%.

To reduce the deviation between the mass fractions of maximum deltaT (0.1 and 0.01 vs 0.0001), the *nCorrectors* of the PIMPLE solver is increased to 5, and it can be observed from figures 3.19 and 3.20 that the deviation is less and and the obtained differences are 0.11% and 0.30%.

The Courant number and deltaT independence study shows us that the PIMPLE solver provides us the convenience of using high Courant number and longer time steps for the pot furnace simulations. It also captures the physics of the simulation properly. Hence, such a setting for the maximum Courant number and maximum deltaT are chosen to run the pot furnace simulations.

4
Results and discussion

In this chapter, various results from the experiment and the simulations are compared, and observations are discussed.

Before moving into these discussions and their derived conclusions, an overview of the simplifications or limiting factors described in previous sections are summarised below.

- Initially, the meshing with a roster plate is replaced with a porous zone as discussed in section 3.5.
- The input boundary conditions for the simulations are averaged values taken from the experiment at an identical location in both experiment and the simulations.
- The source of hot gas to the experiment is through combustion, but to simplify the simulation, the gas composition after the combustion is given as the input to initially study a simpler geometry without the combustion model.
- The pellet positions in the experiment are random but in the simulations, the pellet positions have a pattern and their positioning is explained in section 3.7.
- The simulations are performed with one-way coupling i.e the effect of the fluid on the pellets is present but the pellets do not affect the momentum of the flow field. One-way coupling is chosen because of computational simplicity and as a proof of concept.

The simulation on the numerical model of pot furnace is performed for two different cases. The difference between the two cases is that different MUCS properties are used. The properties for case 1 is taken from [20] and for case 2 it is taken from [22]. Their respective values are tabulated below.

Table 4.1: Activation energy for simulation cases

Activation energy	Case 1	Case 2
Ea (J/kmol)	1.651e5	1e8

From table 4.1 it can observed that the activation energy in case 2 is much higher than case 1. Also from equation 2.23, we could observe that an increase in activation energy leads to lower rate of reaction. The duration or the trend of the conversion process in the simulations and experiment can be discussed by analysing the O_2 volume percentage and the mass fraction conversion plots. Except for the difference

in activation energies, the boundary conditions and physical properties in case 1 and 2 are similar as discussed in section 3.6. The values measured in the experiment using sensors are compared with data obtained by using *extract location* and *plot over time* filter available in ParaView from the simulations. Also the simulation data is recorded from 10 seconds until 1390 seconds with a 10 second interval. The locations extracted in the simulations are from identical locations as in the experiment. The locations are tabulated and represented in the table 4.2 and figure 4.1. The measurements are in Cartesian coordinates with the origin placed as in figure 3.11.

Table 4.2: Locations of sensors in the domain

	Coordinates			
Location ID	X	Y	Z	
L1	0	0	0.034	Sensor Temperature inside Domain
L2	0.122	0	0.035	
L3	0.122	0	0.068	
L4	0.122	0	0.117	
L5	0.122	0	0.228	
L6	0.122	0	0.317	
L7	0.122	0	0.427	
L8	0.122	0	-0.03	Temperature at location of roster plate
L9	0.122	0	0.699	Temperature ahead of the pellets
L10	0.122	0	-0.183	Temperature behind the pellets
L11	0	0	-0.233	O2 volume % sensor
L12	0	0	0.649	O2 volume % sensor

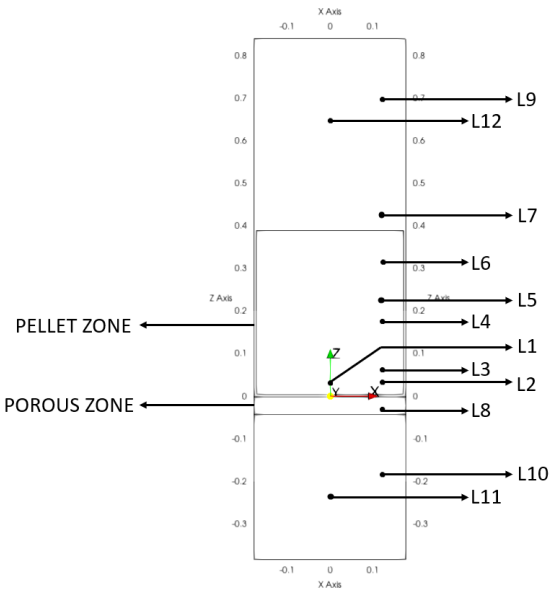


Figure 4.1: Locations of sensors in domain.

4.1 Oxygen volume percentage

The following figures gives the comparison between the experimental and simulation results of the O_2 volume percentage at locations L11 and L12. The supply of oxygen can be split into three regions over the time, where in region 1, 2 and 3 the O_2 percentages are 19%, 1.5% and 42.8% respectively.

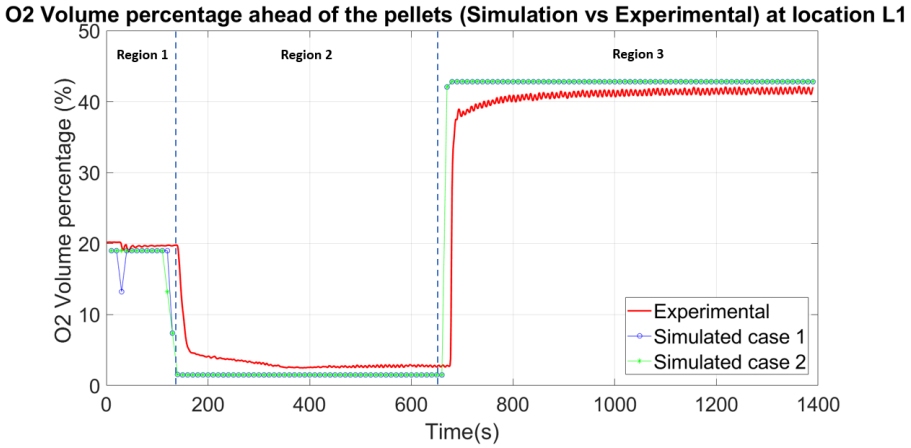


Figure 4.2: Experimental vs simulation O2 volume % ahead of the pellets at location L12.

The similarity in curves for the simulation cases is observed in figure 4.2 because it represents the volume percentage before the gas has reached the pellet bed. Here there is no consumption of oxygen.

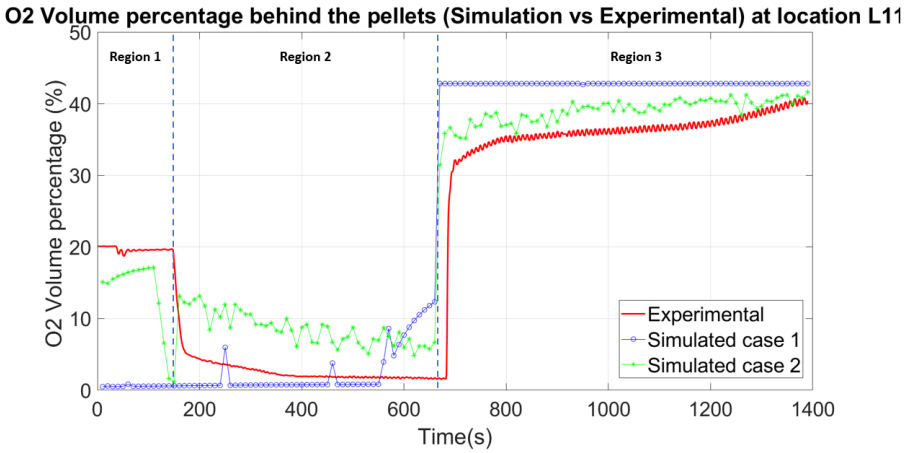


Figure 4.3: Experimental vs simulation O2 volume % behind the pellets at location L11.

Figure 4.3 gives the comparison of the oxygen volume percentage behind the pellets for the simulation cases 1 and 2 compared with the experimental data. It can be observed that the oxygen volume percentage for the simulation case 2 is always

higher than that of the experiment. For simulation case 1, during the first 500 seconds, the oxygen volume percentage is lower than both simulation case 2 and the experiment as the consumption of oxygen is rapid within the regions 1 and 2. After 500 seconds the consumption is higher than the rest of the two cases, as complete conversion would have been achieved. No fluctuations are observed in case 1 due to the lower activation energy and rapid consumption. For simulation case 2, large fluctuations can be observed in figure 4.3. This could be due to the fact that the reaction of the pellets is slow because of a higher activation energy and the oxygen consumption by the pellets is not uniform. Also the measurement is taken at a single point rather than computing the mean flow. The correlation of the curves in simulation case 2 has a similar trend to that of the experiment across all the three regions.

4.2 Mass fractions

The mass fractions are plotted against time for the conversion process of magnetite to hematite. The data for the plots is computed by taking the average of mass fractions of all the pellets at each time step.

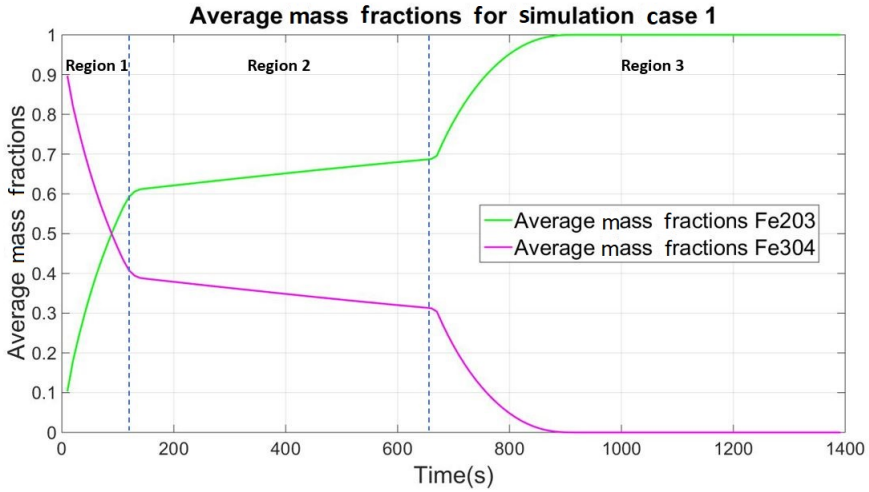


Figure 4.4: Average mass fractions for simulation case 1.

A greater part of the conversion process occurs within the first region of the simulation for the simulated case 1 in figure 4.4. This conversion occurrence is an outcome of the presence of a higher amount of oxygen (shown in figure 4.1) in the domain. This largely contributes to the significant amount of conversion of Fe_3O_4 to Fe_2O_3 within the initial stages of the simulation process itself. Then as the oxygen volume percentage reduces on moving to region 2, the rate of conversion reduces as well, taking a longer time for the conversion as indicated by the slope in region 2. Moving into region 3 with much higher amount of oxygen present in the domain than region 2, the rest of the conversion process occurs quickly, yielding 100% mass fraction of Fe_2O_3.

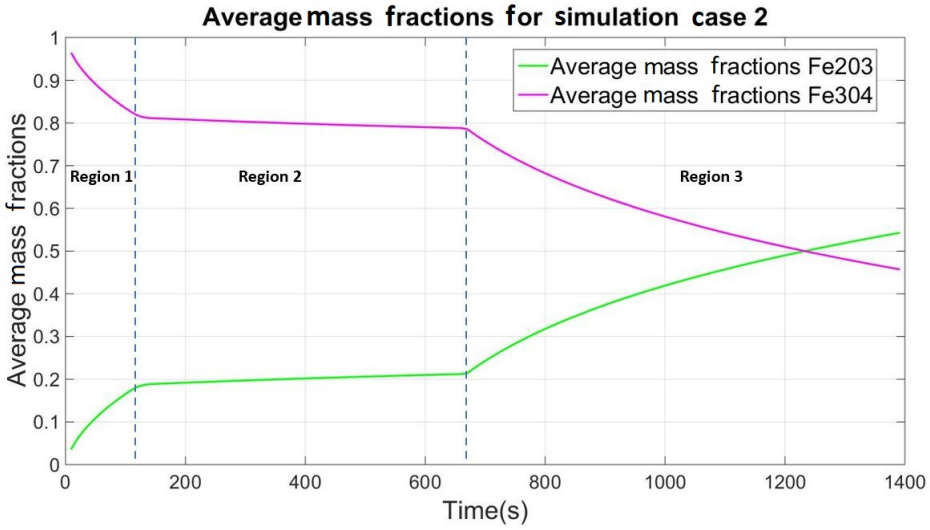


Figure 4.5: Average mass fractions for simulation case 2.

On observing figure 4.5 for the simulated case 2, about only half of the overall conversion has occurred because of the higher activation energy (Ea) compared to that of case 1. A higher activation energy leads to a lower rate of the reaction. Hence, for case 2 with higher Ea, we can observe that the reaction rate has been slowed down to the extent of only 54% of the overall conversion. Even though the conversion has not occurred completely, most of the conversion is still occurring in region 1 and 3 when compared to region 2, because of higher oxygen supply in those regions comparing region 2. Hence, the conversion largely depends on the supply of oxygen.

Table 4.3: Mass gain percentage

	Mass gain %
Experiment	3
Case 1	2.799
Case 2	1.492

From table 4.3, we can observe the mass gain percentage of the pellets after the conversion process. As the conversion process occurs, there will be a consumption of oxygen leading to an increase in overall mass of the pellets. As complete conversion has occurred in the simulated case 1, the overall mass gain is higher in case 1, of value 2.799% which is closer to the value of the experimental mass gain of 3%.

4.3 Temperature

This section will contain results of the comparison between experimental and simulation temperature values at different locations.

4.3.1 Temperature ahead and behind Pellets

The temperature at locations ahead and behind the pellets are measured in the experiment using a suction pyrometer at locations L9 and L10, and compared with the post-processed simulation data.

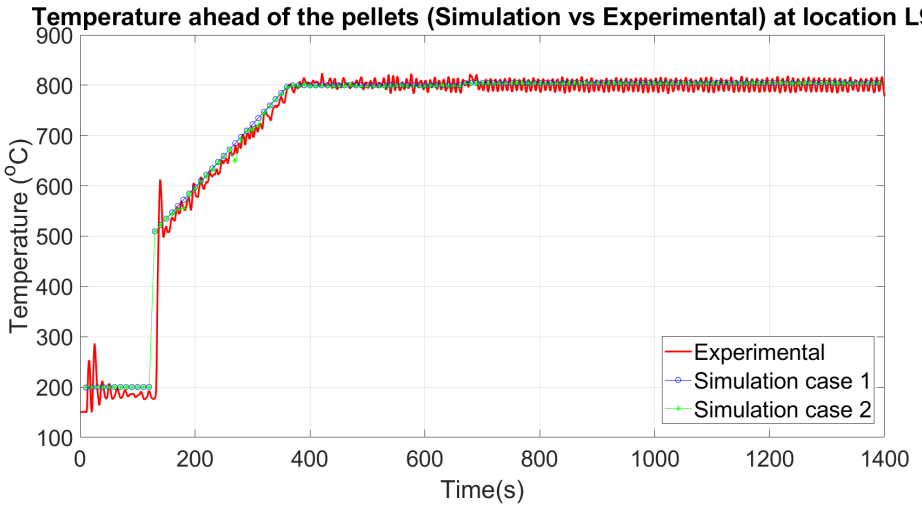


Figure 4.6: Experimental vs simulation temperature at location L9.

The trends in the temperature measured ahead of the pellets in figure 4.6 is similar in both the simulations and the experiment because the gas has not yet reached the pellets at location L9. Since the measurement at location L9 is closer to the inlet, the input temperature profile correlates well with the experiment.

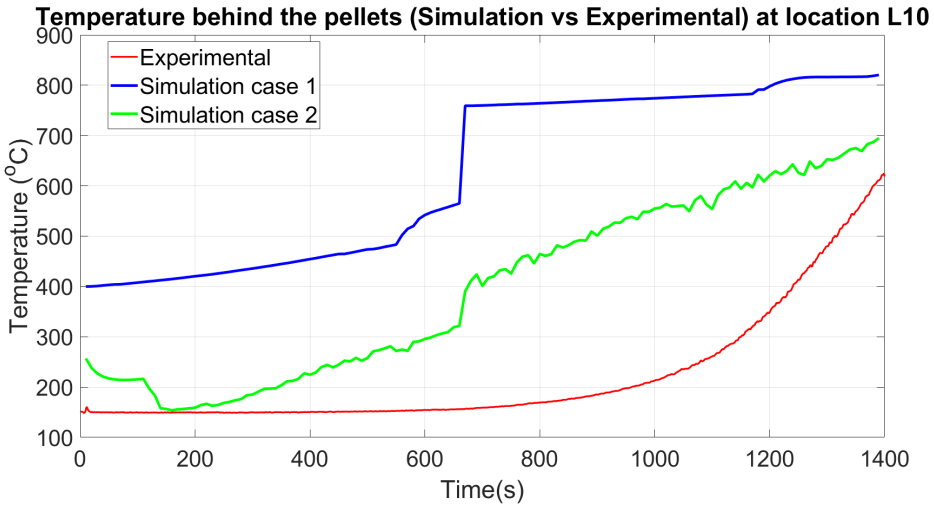


Figure 4.7: Experimental vs simulation temperature at location L10.

A larger deviation in the temperature behind the pellets can be observed in figure 4.7 at L10. It can be observed that, the temperature in the simulation case 2 is lower compared to the simulation case 1 due to the fact that, there is a lesser conversion of Fe_3O_4 to Fe_2O_3 during the initial phase of the simulation. The conversion process

is occurring gradually (Figure 4.5), leading to a gradual increase in temperature. We can observe that for the experiment as illustrated in figure 4.7, that the temperature is lower for the first 800 seconds and increases exponentially. This is due to the fact that the supplied heat will be absorbed by the pellets leading to low temperature in this location behind the pellets. After 800 seconds the pellets would have absorbed energy equal to their activation energy and started to undergo conversion. Since the reaction is exothermic, the temperature in the flow increases. In case 1, within 10 seconds, 10% of the conversion has occurred (Figure 4.4), leading to the flow reaching a temperature of $400°C$ and keeps on increasing. This is because the low activation energy has led to a quick conversion of pellets. We can also say that in the simulations, time taken by the pellets to heat up is not being taken into account. Additionally, the energy absorption is instantaneous.

4.3.2 Temperature within the region of pellets

The temperature in the region where pellets are present are measured using thermocouple sensors from locations L1 to L7 in the experimental setup and the temperature in identical locations are extracted from the simulation data in cases 1 and 2. They are plotted against each other for every location and are shown below.

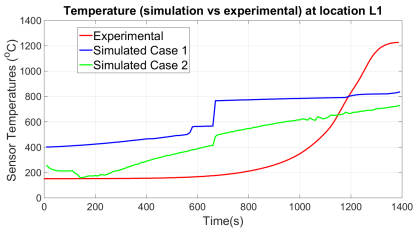


Figure 4.8: Temperature at L1.

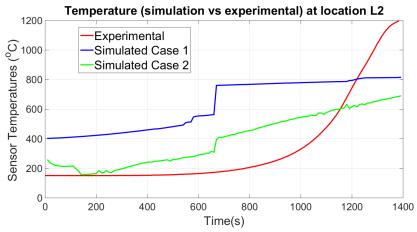


Figure 4.9: Temperature at L2.

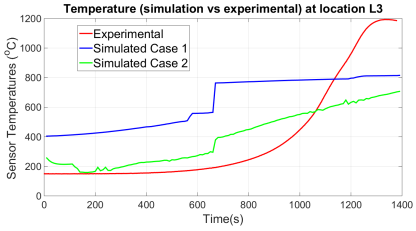


Figure 4.10: Temperature at L3.

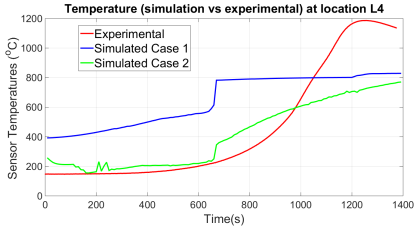


Figure 4.11: Temperature at L4.

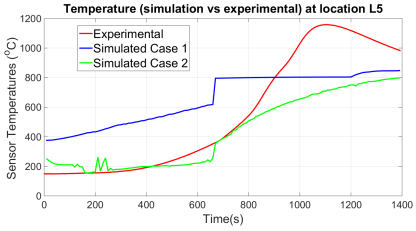


Figure 4.12: Temperature at L5.

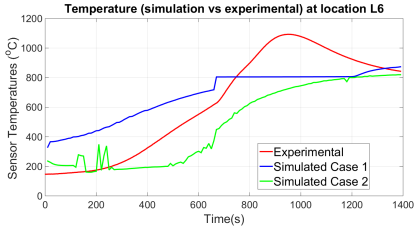


Figure 4.13: Temperature at L6.

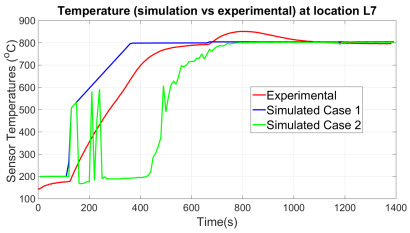


Figure 4.14: Temperature at L7.

The reason for a higher temperature in the simulated case 1 is again accounted by the activation energy. Simulation case 1 has a lower activation energy leading to a higher reaction rate and thus producing a higher temperature from the exothermic reaction. This can be observed in every plot for the locations L1 to L7, where the temperature in simulation case 1 is higher than that of the experiment within the first 600 seconds, whereas for the simulation case 2, the temperature is low unlike case 1 and similar to experiment at certain locations. This lower temperature is due to a higher activation energy used and also occurrence of lesser overall conversion (from section 4.2). The activation energy plays a crucial role in determining the rates of conversions that in turn affects the temperatures in the region of the pellets.

4.3.3 Roster plate/Porous zone temperature

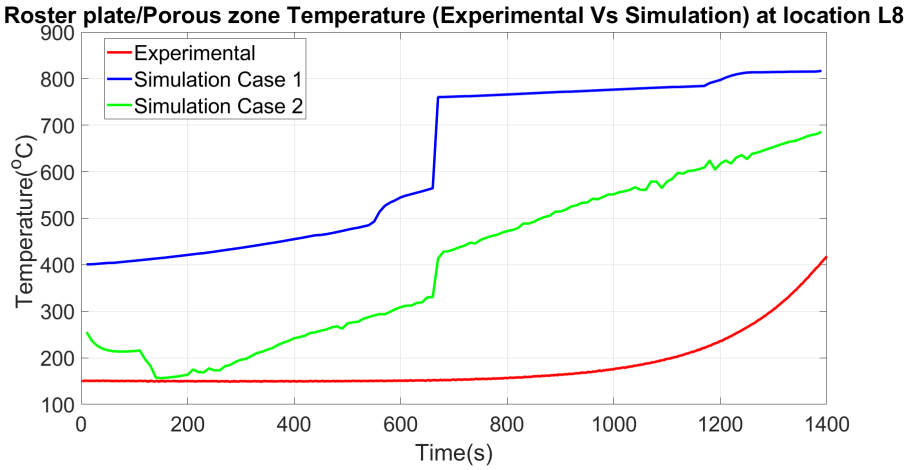


Figure 4.15: Experimental vs simulation temperature at the location of roster plate, L8.

Figure 4.15 illustrates the comparison between the temperature of the roster plate from that of the experiment and the porous zone in the simulations at location L8. The difference is largely because of the usage of a different physical setup i.e the roster plate and the porous zone. This shows that implementing the porous zone instead of the roster plate influences the physics to a noticeable extent, but some trade-offs are needed to come up even with a simplified numerical model to arrive at a conclusive idea on the behaviour.

5
Conclusion

A good correlation in the verification process has helped in concluding the fact that the solver has worked accurately. It is also worthy to mention that the injection model type used in the solver has been widely responsible for particle injections in the form of pellets in the pot furnace simulation as well.

A lagrangian solver with a roster plate makes it difficult for the process of meshing, as we need to consider the cell size to be larger than the particle sizes. Due to large cell sizes, capturing the flow near intricate geometrical areas of the roster that connects the inlet and outlet cylinders is made difficult.

Replacing the roster with a porous zone yields results for validation with the experimental data for pot furnace, which isn't a very good correlation obtained with this process for certain data compared due to the fact that we have avoided the exactness of mimicking the experimental setup. Whereas mimicking the setup (usage of the roster) leads to very low time-stepping (deltaT), making the actual simulation time very high and thus requiring high computational time and effort. Hence, further tuning of the MUCS coefficients are necessary to limit the increase in initial temperatures.

By changing the Courant number and deltaT, a conclusion can be drawn on the fact that the PIMPLE solver allows the usage of a higher value of Courant number and a lower deltaT but with more correctors instead of a lower Courant number and high deltaT that eventually leads to large simulation time.

Activation energy plays a vital role in determining the rate of the conversion process, that leads to differences in temperatures in simulations compared to that of the experiment. The simulation case 1 has recorded a higher temperature and conversion from the initial stages itself because of lower activation energy, but simulation case 2 gives a good correlation of temperature and oxygen volume percentage, whereas the activation energy is not sufficient for a complete conversion of magnetite to hematite for the simulated time (1390 seconds).

Collective usage of such conditions/parameters such as Courant number, deltaT values with PIMPLE solver correctors, type of injection model and a porous zone implementation has aided in obtaining results that will be put to use in the future to generate a numerical model of the exact setup.

6

Recommendations for future work

Obtaining a mesh with the roster plate and being able to simulate it with the aid of computationally heavy resources (computational memory and cores) could result in a very good validation between the simulated and experimental data.

Instead of performing a multi-phase simulation, a single-phase simulation with mesh around the particle zones can be generated, and further, wall fluxes across every particle boundary can be calculated to determine the mass fractions of conversion for Fe_3O_4 to Fe_2O_3 in the stage of post-processing.

The effect of the pellet forces on the fluid (two-way coupling) could be considered and analyzed. It could lead to a more practical approach to problem-solving that was not considered in this thesis, as the main scope of interest revolved around the behaviour of the conversion process and simplicity.

The input data given are from the experiment after the combustion process has occurred, whereas a mesh for the pot furnace with the detailed geometry inclusive of the combustion model can prove to be a vital approach to obtain better correlated results. A model with a switchable fuel input can also be implemented on the detailed numerical model and further, experimentation of different fuels can be done, which will yield results specific to the kind of input used. Such switchable fuel inputs can greatly reduce long experiment hours and the amount of labour.

More realistic particle positioning can be achieved using hexagonal close packing or tetrahedron sphere packing of spheres in the cylinder. Another suggestion is to perform a DEM simulation by injecting the required amount of particles into the cylinder until all particles come into rest and using the positions extracted from the simulation. As the roster plate acts as a physical blockage, the gravity effects could be turned on and a realistic positioning could still be achieved which is not possible with a porous zone as the gravity effect causes the pellets to leave the domain.

Bibliography

[1] LKAB Mining,
www.lkab.com/en/about-lkab/from-mine-to-port/mining/

[2] LKAB Processing,
www.lkab.com/en/about-lkab/from-mine-to-port/processing/

[3] Davidson, L. (2015). Fluid mechanics, turbulent flow and turbulence modeling.

[4] Reynolds number - Definition & regime. www.thermal-engineering.org/what-is-reynolds-number-for-turbulent-flow-definition/

[5] 'An introduction to turbulence and its measurement' , Bradshaw, P. ; Pergamon Press ; 1971 ; ISBN 0080166210

[6] Seyyedvalilu, Hatef. (2018). Re: Difference between standard and realizable k-epsilon model?. Retrieved from: www.researchgate.net/post/Difference-between-standard-and-realizable-k-epsilon-model/5bd21ce04921ee877d74a6da/citation/download

[7] Menter, F. R. (August 1994). "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications". AIAA Journal. 32 (8): 1598–1605. https://arc.aiaa.org/doi/10.2514/3.12149.

[8] Papanastassiou, D., Bitsianes, G. Modelling of heterogeneous gas-solid reactions. Metall Mater Trans B 4, 477–486 (1973). https://doi.org/10.1007/BF02648700

[9] Szűcs, T., Szentannai, P. Determining the mass-related reaction effectiveness factor of large, nonspherical fuel particles for bridging between intrinsic and apparent combustion kinetics. J Therm Anal Calorim 141, 797–806 (2020). https://doi.org/10.1007/s10973-019-09085-9

[10] Çengel, Yunus A. (2003) Heat transfer : a practical approach.McGraw-Hill series in mechanical engineering, ISBN = 0072458933

[11] Thamali R. Jayawickrama, Nils Erland L. Haugen, Matthaus U. Babler, M.A. Chishty, Kentaro Umeki, The effect of Stefan flow on Nusselt num-

ber and drag coefficient of spherical particles in non-isothermal gas flow, International Journal of Multiphase Flow, Volume 140, 2021, 103650, ISSN 0301-9322,https://doi.org/10.1016/j.ijmultiphaseflow.2021.103650.

[12] H. Versteeg, W. Malalasekera, An Introduction to Computational Fluid Dynamics, 2nd Edition, Pearson Education Limited, Harlow, England, 2007.

[13] www.openfoam.com/documentation/guides/latest/doc/guide-fos-field-CourantNo.html

[14] Crowe, Clayton and Schwarzkopf, John and Sommerfeld, Martin and Tsuji, Yutaka. (2011) Multiphase flows with droplets and particles, isbn = 1439840504

[15] Haukur Elvar Hafsteinsson (2009) 'Porous media in openFOAM'. www.tfd.chalmers.se/~hani/kurser/OS_CFD_2008/HaukurElvarHafsteinsson\/haukurReport.pdf

[16] https://www.simscale.com/forum/t/cfd-pimple-algorithm/81418

[17] openfoam,title=2.1 File structure of OpenFOAM cases www.openfoam.com/documentation/user-guide/2-openfoam-cases/2.1-file-structure-of-openfoam-cases, journal=OpenFOAM

[18] www.openfoam.com/documentation/guides/latest/man/reactingHeterogenousParcelFoam.html

[19] OpenFOAM USER GUIDE v2012, https://www.openfoam.com/documentation/guides/latest/doc/index.html

[20] Papanastassiou, D., Bitsianes, G. Mechanisms and kinetics underlying the oxidation of magnetite in the induration of iron ore pellets. Metall Mater Trans B 4, 487–496 (1973). https://doi.org/10.1007/BF02648701

[21] http://openfoamwiki.net/index.php/Sig_Turbomachinery_/_ERCOFTAC_conical_diffuser

[22] Monazam, Esmail Breault, Ronald Siriwardane, Ranjani. (2014). Kinetics of Magnetite (Fe3O4) Oxidation to Hematite (Fe2O3) in Air for Chemical Looping Combustion.

A

Appendix 1

A.1 Solver verification - Case files

A.1.1 reactingCloud1Properties

```
/*---------------------------------*- C++ -*----------*\
| =========                       |                   |
| \\      /  Field                | OpenFOAM: The Open Source CFD Toolbox
|  \\    /   Operation            | Version:  v2012
|   \\  /    And                  | Website:  www.openfoam.com
|    \\/     Manipulation         |
\*---------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      reactingCloud1Properties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

solution
{
    active          yes;
    coupled         true;
    transient       yes;

    maxCo           0.3;
    cellValueSourceCorrection off;


    sourceTerms
    {
        resetOnStartup  false;
        schemes
        {
```

```
            rho             explicit  1;
            U               explicit  1;
            Yi              explicit  1;
            h               explicit  1;
            radiation       explicit  1;
        }
    }

    interpolationSchemes
    {
        rho             cell;
        U               cellPoint;
        thermo:mu       cell;
        T               cell;
        Cp              cell;
        kappa           cell;
        p               cell;
    }

    integrationSchemes
    {
        U               Euler;
        T               analytical;
    }
}


constantProperties
{
    rho0            5100;
    T0              303;
    Cp0             850;

    hRetentionCoeff 0;
    volumeUpdateMethod  constantVolume;
}


subModels
{
    particleForces
    {}

    injectionModels
    {
        model1
```

```
    {
        type            manualInjection;
        massTotal       0;
        parcelBasisType fixed;
        nParticle       1;
        //parcelsPerSecond 8442;
        SOI             0;
        positionsFile   "reactingCloud1Positions";
        U0              (0.02 0 0);
        sizeDistribution
        {
            type        fixedValue;
            fixedValueDistribution
            {
                value       0.011;
            }
        }
    }
}
dispersionModel gradientDispersionRAS;

patchInteractionModel standardWallInteraction;

heatTransferModel RanzMarshall;

compositionModel singleMixtureFraction;

phaseChangeModel none;

stochasticCollisionModel none;

surfaceFilmModel none;

radiation       off;

standardWallInteractionCoeffs
{
    type            rebound;
}

RanzMarshallCoeffs
{
    BirdCorrection  off;
}

heterogeneousReactingModel  MUCSheterogeneousRate;
```

```
    MUCSheterogeneousRateCoeffs
    {
        D12             2.724e-4; //m2/s
        epsilon         0.41;
        gamma           3.07;
        sigma           1;
        E               1;
        A               3.14e4;    // m/s
        Aeff            0.7;
        Ea              1.651e5;   // J/kmol
        O2              O2;

        // nuFuel*Fe3O4 + nuOx*O2 => nuProd*Fe2O3
        nuFuel          2.0;
        nuProd          3.0;
        nuOx            0.5;
        fuel            Fe3O4;
        product         Fe2O3;
    }

    singleMixtureFractionCoeffs
    {
        phases
        (
            gas
            {
            }
            liquid
            {
            }
            solid
            {
                Fe3O4 1;
                Fe2O3 0;
            }
        );
        YGasTot0            0;
        YLiquidTot0         0;
        YSolidTot0          1;
    }
}


cloudFunctions
{
```

```
}


// ****************************************** //
```

A.1.2 reactingCloud1Positions

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       vectorField;
    location    "constant";
    object      reactingCloud1Positions;
}
// * * * * * * * * * * * * * * * * * * * * *  * //
scale 0.01;
(
(0 1 0.5)
(0 2 0.5)
(0 3 0.5)
(0 4 0.5)
(0 5 0.5)
(0 6 0.5)
(0 7 0.5)
(0 8 0.5)
)
// ****************************************** //
```

A.1.3 Turbulence model

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      turbulenceProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * //

simulationType  RAS;

RAS
{
    RASModel            kOmegaSST; // kEpsilon;
```

```
    turbulence          on;

    printCoeffs         on;
}


//
```

A.1.4 Thermophysical properties

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      thermophysicalProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * *  * //

thermoType
{
    type            heRhoThermo;
    mixture         reactingMixture;
    transport       sutherland;
    thermo          janaf;
    energy          sensibleEnthalpy;
    equationOfState perfectGas;
    specie          specie;
}

chemistryReader  foamChemistryReader;

foamChemistryFile "<constant>/foam.inp";

foamChemistryThermoFile "<constant>/foam.dat";

inertSpecie    N2;

liquids
{}

solids
{
    Fe3O4
```

```
    {
        defaultCoeffs   no;
        Fe3O4Coeffs
        {
            rho             5100;
            Cp              850;
            kappa           0.04;
            Hf              0;
            emissivity      1.0;
            W               232;
        }
    }

    Fe2O3
    {
        defaultCoeffs   no;
        Fe2O3Coeffs
        {
            rho             5100;
            Cp              710;
            kappa           0.04;
            Hf              525e3; //Heat of reaction HfProd - HfFuel
            emissivity      1.0;
            W               159;   //Kg/Kmole
        }
    }
}



//
```

B

Appendix 2

B.1 Simulation - Case files

B.1.1 Initial time directory - 0

B.1.1.1 Temperature - T

```
/*---------------*- C++ -*----------*\
| =========                 |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|  \\    /   O peration     | Version:  v2012
|   \\  /    A nd           | Website:  www.openfoam.com
|    \\/     M anipulation  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      T;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 0 0 1 0 0 0];

internalField   uniform 423;

boundaryField
{
    walls
    {
        type            zeroGradient;
    }
    porosityWall
    {
        type            zeroGradient;
```

```
    }
    outlet
    {
        type            inletOutlet;
        inletValue      $internalField;
    }
    inlet
    {
        type            uniformFixedValue;
        uniformValue    table
        (
        (0 473)
        (121 473)
        (122 773)
        (362 1073)
        (1390 1073)
        );

    }
}


// ************************************************* //
```

B.1.1.2 Velocity - U

```
dimensions      [0 1 -1 0 0 0 0];


internalField   uniform (0 0 0);

boundaryField
{
    inlet
    {
        type            uniformFixedValue;
        uniformValue table
        (
        (0 (0 0 -2.965))
        (10 (0 0 -2.965))
        (20 (0 0 -1.89))
        (147 (0 0 -1.89))
        (152 (0 0 -1.3))
        (405 (0 0 -2.1))
        (1390 (0 0 -2.1))
        );
```

X

```
    }
    outlet
    {
        type            pressureInletOutletVelocity;
        value            $internalField;
    }
    walls
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
    porosityWall
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }
}
```

B.1.1.3 Oxygen - O_2

```
dimensions      [0 0 0 0 0 0 0];

internalField   uniform 0.2;

boundaryField
{
    walls
    {
        type            zeroGradient;
    }
    porosityWall
    {
        type            zeroGradient;
    }
    outlet
    {
        type            inletOutlet;
        inletValue      $internalField;//uniform 0.2;
        value           $internalField;//uniform 0.2;
    }
    inlet
    {
        type            uniformFixedValue;
        uniformValue    table
        (
         (0   0.19)
```

```
            (110   0.19)
            (140   0.015)
            (665   0.015)
            (670   0.428)
            (1390   0.428)
           );
    }
}
```

B.1.2 Physical setup - Constant

B.1.2.1 fvOptions

```
/*---------------------------------*- C++ -*----------
| =========                       |
| \\      /  F ield               | OpenFOAM: The Open Source CFD Toolbox
|  \\    /   O peration           | Version:  2.2.0
|   \\  /    A nd                 | Web:      www.OpenFOAM.org
|    \\/     M anipulation        |
\*---------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      fvOptions;
}
// * * * * * * * * * * * * * * * * * * * * * * * * ** * //

limitT
{
    type            limitTemperature;
    active          yes;

    selectionMode   all;
    min             278;
    max             3000;
}

porosity1
{
    type            explicitPorositySource;

    explicitPorositySourceCoeffs
    {
```

```
        selectionMode   cellZone;
        cellZone        porosity;

        type            DarcyForchheimer;

        d   (1.35e7 1.35e7 1.35e7);
        f   (0 0 0);

        coordinateSystem
        {
            origin  (0 0 0);
            e1      (1 0 0);
            e2      (0 1 0);
        }
    }
}


// ************************************************* //
```

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY