FMH606 Master's Thesis 2021

Process Technology

# Simulation of hydrogen tank refueling

Jose Luis Salcido Sanchez

Faculty of Technology, Natural sciences and Maritime Sciences

Campus Porsgrunn

# University of South-Eastern Norway

www.usn.no

**Course**: FMH606 Master's Thesis, 2021

**Title**: Simulation of hydrogen tank refueling

**Number of pages**: 98

**Keywords**: CFD, Simulation, Hydrogen, Tank refueling, Heat transfer coefficient.


| | |
|---|---|
| **Student:** | José Luis Salcido Sanchez |
| **Supervisor:** | Knut Vågsæther |
| **External partner:** | H2Maritime, IFE |

**Summary:**

Several simulations of hydrogen filling process for a type IV tank were completed using openFOAM Computational Fluid Dynamics (CFD) open-source code. CFD codes have been proven to be a useful method for estimating temperature and pressure distribution developed inside the tank during refueling operations. All simulations were completed in a 2-D mesh and for a simulated time of 20 seconds with varying initial tank pressures and mass flow rate at the inlet. The transient, compressible and heat transfer rhoCentralFoam solver was used along with the k-ω SST turbulence model to reproduce the effects of compressibility of the gas during the filling.

The simulation cases were designated into two groups for comparing a 10 to 50 g/s mass flow rate, along with different initial tank pressures of 1, 5, 10, 20, 30 and 35 MPa. The gas inlet temperature was 273.15 K, with an initial tank and walls temperature of 298.15 K. Surrounding temperature was selected as 298.15 K with a constant convective transfer coefficient of 10 W/m² K. The simulated wall thickness is 5 mm with a thermal conductivity of 0.4 W/m K for simulating a polymer liner.

Results for the temperature and pressure distribution within the tank are in agreement with previous research papers' findings for which larger initial pressures lead to minor increments in temperature during refueling. Larger initial pressures resulted in less gas velocities at the inlet. Temperature rise is heavily dependent of the inlet gas mass flow, as a higher flow will result in more mass dispensed into the tank. The applicability of these findings suggests that at the start of the gas refueling process, a low mass flow rate can be dispensed, as the instant change in temperature is smaller, and once a higher pressure is developed within the tank, increase the mass flow to achieve a faster fill.

The convective heat transfer coefficient is influenced by the mass flow rate. For lower initial pressure simulations, higher values are achieved at the end region of the tank, far away from the inlet, due to gas compression. In some instances, negative coefficient values were developed, as tank wall temperature went lower than initial temperature condition, leading to a negative heat flux into the tank.

Higher initial pressure cases resulted in oscillations of the variables through the simulation. Finer mesh analysis is suggested to verify mesh dependency.

# Preface

This thesis aims to evaluate the refueling process of a hydrogen type IV tank for maritime applications and the development of corresponding variables such as pressure and temperature distribution as well as heat transfer. In order to do it, I performed several CFD simulations using openFOAM open-source code, matching different inlet mass flows with different initial pressures. The goal was to assess whether CFD method can provide useful insight into a real application.

I would like to thank anyone who helped me along the way, even if they were not directly involved in the project.

Porsgrunn, 17th of May of 2021

José Luis Salcido Sanchez

# Contents

# Table of Figures

# Table of Tables

# Nomenclature

$C_p$     heat capacity at constant pressure [J/K]

$C_v$     heat capacity at constant volume [J/K]

$C_\mu$     turbulent viscosity [Kg/m.s]

div     divergence

E     energy flux

grad     gradient

h     convective heat transfer coefficient [W/m$^2$.K]

I     turbulent intensity

k     turbulent kinetic energy [m$^2$/s$^2$]

L     reference scale length [m]

n     moles [mol]

p     pressure [Pa]

**P**     mean pressure [Pa]

p'     pressure fluctuation [Pa]

$P_c$     critical pressure [Pa]

q"     heat flux [W/m$^2$]

R     gas constant [J/mol.K]

Re     Reynolds number

$S_M$     source term

t     time [s]

T     temperature [K]

$T_\infty$     surrounding temperature [K]

$T_c$     critical temperature [K]

$T_r$     reduced temperature [K]

$T_w$     wall temperature [K]

u     x velocity [m/s]

**u**     u vector

**U**     mean x velocity [m/s]

u'     x velocity fluctuation [m/s]

v     y velocity [m/s]

V     volume [m$^3$]

**V**     mean y velocity [m/s]

v'     y velocity fluctuation [m/s]

$V_c$     critical volume [m$^3$]

$V_m$     molar volume [m$^3$/mol]

w     z velocity [m/s]

**W**     mean z velocity [m/s]

w'     z velocity fluctuation [m/s]

$\Delta t$     time step [s]

$\kappa$     thermal conductivity [W/m.K]

$\mu$     dynamic viscosity [Kg/m.s]

$\rho$     density [kg/m$^3$]

$\tau$     shear stress [N/m$^2$]

$\Phi$     dissipation function

**ω**     acentric factor

ω     specific rate of dissipation [s$^{-1}$]

# 1 Introduction

This chapter provides an overview of the thesis's context. The thesis' objective is explained and previous work that is relevant to the subject is discussed.

## 1.1 Background

From several years until now, the world has chosen to move to carbon neutrality with the goal to reduce carbon footprint. Hydrogen has been recently become a fuel of interest, as it provides carbon free energy for transportation. Commercial vehicles that operate using hydrogen have been in production in recent years, so couple to that, several research have been performed on smaller tanks, to try to understand the pressure and temperature field developed inside the hydrogen tank as it is being refueled. Land vehicle hydrogen fuel tanks must comply to the maximum wall temperature of 85 °C specified in the ISO/TS 15869 international standard.

The scope of this research involves the analysis and computer simulation of a type IV tank intended for maritime transportation. The size of the simulated tank is bigger in comparison to previous studies for land vehicles, with a diameter of 0.7 meters and a length of 6.6 meters. The tank is being manufactured by UMOE Advance Composites and their maximum inside wall temperature specified is 60 °C as compared to the ISO/TS 15869 standard. This presents an issue, as lower mass flow inlet rates must be used to control the increase of temperature during refilling.

Computational fluid dynamics (CFD) has proved to be an extremely useful tool for the simulation and validation of results. Several papers have demonstrated acceptable agreement between the computational model and experimental testing data, as well as its feasibility to be use for preliminary testing for design and optimization. Due to this, the "Open-source Field Operation and Manipulation" (openFOAM) software for CFD was used for the work on this thesis.

## 1.2 Previous work

Extensive literature research was performed to achieve a better understanding of the hydrogen tank refilling process phenomena.

de Miguel et al [1] studied the influence of the inlet gas temperature and how it affects the average tank temperature at the end of filling process for a type IV tank. Their findings, validated by comparing to experimental tests data, show that a preconditioning or reduction of the inlet gas temperature to around -40 ° C, greatly helps in the reduction of the final average tank temperature as compared to a 0 ° C inlet gas temperature.

Galassi et al [2] performed CFD simulations for fast filling processes for a 70 MPa hydrogen type IV tank, using ANSYS CFX. Their results indicate that higher inlet pressure rates lead higher changes in temperature within the tank. While also, dispensing a colder hydrogen gas will result in lower gas and wall temperatures below the maximum temperature established by internal regulation standards.

Hosseini et al [3] studied the effect of the initial gas tank pressure on the average final temperature of the filling process by thermodynamic modelling. Their study shows that a lower initial pressure at the tank, will lead to rapid increase in gas temperature during the first seconds of the filling process due to high pressure drop of the inlet gas, and due to the

negative Joule-Thomson coefficient for hydrogen gas. On the other hand, a higher initial tank pressure will lead to a lower increase in the average tank temperature during the first seconds of the process, but with a lower quantity of dispensed gas into the tank.

Heitsch et al [4] performed CFD simulations on type III and type IV tanks using ANSYS CFX software with two different initial pressures of 100 and 200 bar. Their findings are aligned to previous researcher papers, in which a higher initial tank pressure results in a lower rise in temperature during the whole refueling process.

Dicken, C. and W. Mérida [5] performed asymmetric 2-D simulations using ANSYS Fluent software, on a type III tank with the goal to better understand the wall heat transfer and convective coefficient. Their findings show that the convective coefficient is heavily dependent of the inlet mass flow rate, and maximum heat transfer within the first seconds of the filling process. Heat transfer and local convective coefficient are at the highest at the back end of the tank.

Bourgeois et al [6] obtained experimental and modeled data from a type IV composite tank. They studied the heat transfer coefficient between inner wall and gas with two different mass flow rates. Their experimental data indicates that the heat transfer coefficient is heavily influenced by the gas mass flow, as a higher mass flow rate results in greater heat transfer, thus a higher heat transfer coefficient. Also, the convective heat transfer coefficient increased rapidly during the first seconds of refueling, to later decline slowly, almost constant for the rest of the filling.

Tagaki et al [7] performed 1-D and 3-D simulations using openFOAM code. Their findings for the 1-D analysis underpredicted the temperature field developed within the tank. While the 3-D analysis over predicted the temperature field as compared to experimental data. The results also show that a rise in temperature is located and developed on the filled region of the tank, opposite to the inlet.

Lizka et al [8] performed CFD modelling using ANSYS Fluent software on tanks with aluminum and phase change materials (PCM) liners. Their model was able to predict the tank pressure and temperature. The PCM liner simulations were characterized by a cooler region at the end of the tank cylinder, as the liner was not able to track the gas temperature in that region. For real hydrogen refueling operations, this effect would not be expected.

## 1.3 Objective

The objective of this thesis is to obtain a further insight into the hydrogen tank refueling process of a prototype type IV tank using CFD. The goal is to obtain and evaluate the temperature and pressure developed during the filling process, as well as the wall-gas heat transfer coefficient at three different regions along the tank. The tank dimensions, mass flow rate and maximum wall temperature requirement were provided by the manufacturer. Since there is no experimental data to compare the computational model, several parameters used were obtained from research literature for type IV tanks, such as liner material thickness and thermal conductivity, as well as inlet diameter and external heat transfer coefficient between tank external wall and environment.

# 2 Theory

## 2.1 The Navier-Stokes Equations

The N-S equations are the governing equations that describe the motion of fluids, which are the fluid representation of Newton's second law stating that the rate of momentum change is proportional to the sum of the forces applied to a body.

### 2.1.1 Continuity Equation

The governing continuity equation is derived by diving the domain into smaller control volumes, where the mass flow into the control volume must be equal to the mass flow out of the control volume, thus achieving mass conservation. This rate of change is expressed by:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} + \frac{\partial (\rho v)}{\partial y} + \frac{\partial (\rho w)}{\partial z} = 0 \qquad (2.1)$$

$$\frac{\partial \rho}{\partial t} + div(\rho \boldsymbol{u}) = 0 \qquad (2.2)$$

### 2.1.2 Momentum Equation

The momentum governing equation takes into consideration the sum of all the forces acting on the control volume for calculating the rate of change of momentum. These forces are pressure, tension, and shear stresses for all three X, Y and Z directions plus additional body forces. This rate of change of momentum is expressed by:

$$\frac{Du}{Dt} \cdot \rho = -\frac{\partial p}{\partial x} + \frac{\partial \tau_{xx}}{\partial x} + \frac{\partial \tau_{yx}}{\partial y} + \frac{\partial \tau_{zx}}{\partial z} + S_{Mx} \qquad (2.3)$$

$$\frac{Dv}{Dt} \cdot \rho = -\frac{\partial p}{\partial y} + \frac{\partial \tau_{xy}}{\partial x} + \frac{\partial \tau_{yy}}{\partial y} + \frac{\partial \tau_{zy}}{\partial z} + S_{My} \qquad (2.4)$$

$$\frac{Dw}{Dt} \cdot \rho = -\frac{\partial p}{\partial z} + \frac{\partial \tau_{xz}}{\partial x} + \frac{\partial \tau_{yz}}{\partial y} + \frac{\partial \tau_{zz}}{\partial z} + S_{Mz} \qquad (2.5)$$

Assuming rotational equilibrium where the shear stresses in the opposite direction are equal; and assuming Newtonian fluid where the viscous forces are proportional to the strain or deformation rate. We get:

$$\frac{\partial (\rho u)}{\partial t} + div(\rho u \boldsymbol{u}) = -\frac{\partial p}{\partial x} + div(\mu \cdot grad\, u) \qquad (2.6)$$

$$\frac{\partial(\rho v)}{\partial t} + div(\rho v \boldsymbol{u}) = -\frac{\partial p}{\partial y} + div\ (\mu \cdot grad\ v) \tag{2.7}$$

$$\frac{\partial(\rho w)}{\partial t} + div(\rho w \boldsymbol{u}) = -\frac{\partial p}{\partial z} + div\ (\mu \cdot grad\ w) \tag{2.8}$$

For this work gravity body force was neglected as simulation time for the different cases was not long enough to account for its effect

### 2.1.3  Energy Equation

The governing energy equation, based on the first law of thermodynamics, states that the rate of increase of energy is equal to the net rate of head added to the control volume plus the net rate of work done on the control volume. And if we extract the changes in kinetic energy it can be rearranged to obtain the energy equation in terms of internal energy as follows [9]:

$$\frac{\partial(\rho i)}{\partial t} + div(\rho i \boldsymbol{u}) = -p\ div\ \boldsymbol{u} + div\ (\kappa\ grad\ T) + \Phi + S_i \tag{2.9}$$

## 2.2  Equation of State

The equation of state (EOS) provides a relationship between the state variables such as Pressure, Volume and Temperature within each other under a specific set of physical conditions. The most widely use equation of state based on the Ideal Gas law states that:

$$pV = nRT \tag{2.10}$$

But for gases at high pressures, the volume occupied by the gas is small and the ideal gas law does not take into account the attraction and repulsive forces shown at high pressures. For this reason, the Peng-Robinson equation of state can account for real gases effects [10]:

$$p = \frac{RT}{V_m - b} - \frac{a\alpha}{V_m^2 + 2bV_m - b^2} \tag{2.11}$$

$$a = 0.45724\ \frac{R^2 T_c^2}{p_c} \tag{2.12}$$

$$b = 0.07780\ \frac{RT_c}{p_c} \tag{2.13}$$

$$\alpha = (1 + \kappa\ (1 - T_r^{1/2}))^2 \tag{2.14}$$

$$\kappa = 0.37464 + 1.54226\omega - 0.26992\boldsymbol{\omega}^2 \tag{2.15}$$

$$T_r = \frac{T}{T_c} \tag{2.16}$$

## 2.3 Heat transfer

Heat transfer is the transit of thermal energy because of a temperature differential through space. There are three modes for heat transfer: conduction, convection, and radiation. For this thesis report the radiation heat transfer effects were neglected.

### 2.3.1 Conduction

Based on references [11] and [9] conduction can be described as the transfer of energy from more energetic to less energetic particles of a material as a result of particle interactions. This means that heat flows from a higher temperature region to a lower temperature region. Heat transfer by conduction can be modeled using rate equations. These equations can be used to calculate energy transfer per unit time, and the rate equation is called the Fourier's Law for heat conduction, expressed for a one-dimensional scenario as:

$$q_x'' = -\kappa \frac{dT}{dx} \qquad (2.17)$$

Where, $q_x''$ is the heat transfer rate per unit time per unit area, refer as heat flux. The parameter $\kappa$ is the material's thermal conductivity. For a homogeneous, isotropic material, thermal conductivity is a positive scalar quantity. And $dT/dx$ is the temperature gradient in the direction.

### 2.3.2 Convection

Convection heat transfer can be described as being composed by two mechanism. First, by conduction as previously discussed, and second, energy transfer by fluid in motion. This fluid motion is synonymous with the idea that vast quantities of molecules are moved together at any given time. In the presence of a temperature gradient, such motion leads to heat transfer.

Convection is a much more complicated heat transfer mode compared to conduction. As it depends on the fluid density, viscosity, thermal conductivity, heat capacity and velocity. Additional factors such as laminar or turbulent flow regime and geometry add to the complexity of this heat transfer mode. It can be expressed by its rate equation, commonly known as Newton's Cooling Law, expressed as:

$$q'' = h(T_w - T_\infty) \qquad (2.18)$$

Where the convective heat flux $q''$, is proportional to the difference between the wall (or surface) and fluid temperatures, $T_w$ and $T_\infty$, respectively. And $h$ is the heat transfer coefficient.

This equation oversimplifies the nature of the heat transfer coefficient, since as previously mentioned, other factors and variables influence its behavior.

Rearranging equation 2.18, we can obtain the heat transfer coefficient as the ratio of the convective heat flux and the temperature difference between the wall and fluid, expressed as:

$$h = \frac{q''}{T_w - T_\infty} \qquad (2.19)$$

From now on, the fluid temperature is referenced as reference temperature. The reference temperature varies from external to internal flow. For external flows, the temperature of the bulk free stream fluid away from the wall can be used as the reference temperature. On the

other hand, for internal flows, a mass flow averaged temperature refer as mean temperature is used. This mean temperature then varies along the flow direction.

## 2.4  Turbulence modelling

Most of the fluid flows encountered in engineering problems involve turbulent behavior above a certain Reynolds number. Thus, it is important to model turbulence effect as the flow properties will vary randomly. There several methods called turbulent models capable of addressing turbulence numerically. The most commonly method used for engineering applications is the Reynolds Averaged Navier-Stokes model (RANS). Other methods exist, such as Large Eddie Simulation (LES) and Direct Numerical Simulation (DNS), which are capable to provide higher accuracy but with an increase in computational cost and time [9].
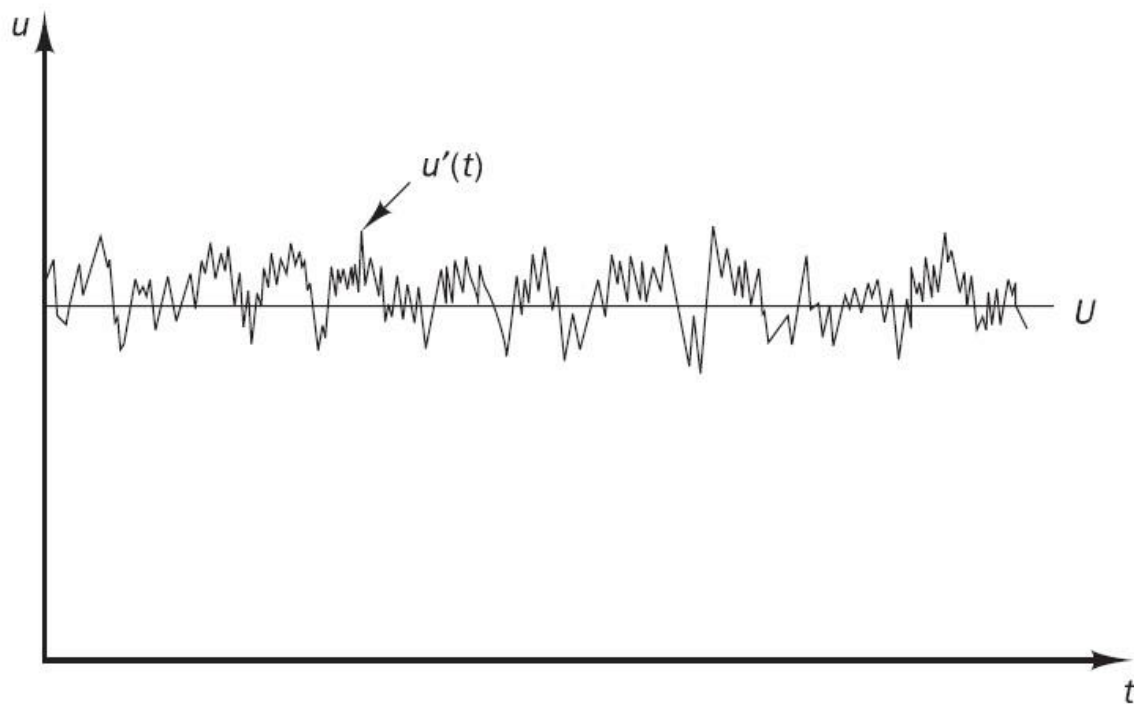


Figure 2.1 Turbulent flow velocity over time

For the work on this thesis the RANS method is used as it provides sufficient accuracy with low computational time. This model is based on the Reynolds decomposition technique in which the velocity and pressure flow properties (*u, p*) can be decompose into its mean and fluctuating value over time as shown in Figure 2.1, so that:

$$u = U + u' \tag{2.20}$$

$$v = V + v' \tag{2.21}$$

$$w = W + w' \tag{2.22}$$

$$p = P + p' \tag{2.23}$$

For which time averaging is applied as:

$$U = \frac{1}{\Delta t} \int_0^{\Delta t} U dt \qquad (2.24)$$

And the time average of a fluctuation is zero:

$$\bar{u}' = \frac{1}{\Delta t} \int_0^{\Delta t} u'(t) dt \equiv 0 \qquad (2.25)$$

Inserting this time average terms into the Navier Stokes equations we get (shown only for the x direction here):

$$\frac{\partial(\bar{\rho}\tilde{U})}{\partial t} + div(\bar{\rho}\tilde{U}\mathbf{\tilde{U}}) = -\frac{\partial \bar{P}}{\partial x} + div(\mu \, grad \, \tilde{U}) + \left[ -\frac{\partial \overline{(\bar{\rho}u'^2)}}{\partial x} - \frac{\partial \overline{(\bar{\rho}u'v')}}{\partial y} - \frac{\partial \overline{(\bar{\rho}u'w')}}{\partial z} \right] + S_{Mx} \qquad (2.26)$$

Using the Boussinesq approach, where the stress is proportional to the strain rate, new stress terms appear denominated Reynold stresses or turbulent stresses:

$$\tau_{ij} = -\rho\overline{u_i'u_j'} = \mu_t \left( \frac{\partial \bar{u}_i}{\partial xj} + \frac{\partial \bar{u}_j}{\partial xi} \right) - \frac{2}{3}\rho k \delta_{ij} \qquad (2.27)$$

Which require additional equations to be solved and to close the system of RANS equations. For this, several turbulence models are used depending on the number of extra transport equations required for the solution.

## 2.4.1  k-ω SST (Shear Stress Transport) model

The k-ω SST turbulence model is a two-equation model and was developed as combination of the k-ε model for the free stream fully turbulent region away from the wall, and the k-ω standard model for the region near the wall. The model introduces two variables, $k$ as the turbulent kinetic energy and $\omega$ as the specific rate of dissipation of the turbulent kinetic energy into thermal energy. In openFOAM, this model introduces new fields which need to be solved, and it is possible to calculate reasonable boundary conditions values as [12]:

$$k = \frac{3}{2}(UI)^2 \qquad (2.28)$$

$$I = 0.16 \, Re^{-\frac{1}{8}} \qquad (2.29)$$

$$\omega = \frac{\sqrt{k}}{C_\mu^{\frac{1}{4}} L} \tag{2.30}$$

Where $C_\mu = 0.09$, $\boldsymbol{U}$ is the mean inlet velocity and $I$ is the turbulent intensity, with a typical value between 1% to 10%. But for this thesis work the Ansys Fluent Equation 2.29 is used [13]. And $L$ is a reference length scale.

The turbulence kinetic energy (k) and dissipation rate (ω) transport equations (shown only for the x direction here) are:

$$\frac{\partial(\rho k)}{\partial t} + \frac{\partial}{\partial x_i}(\rho k \bar{u}_i) = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] + 2\mu_t S_{ij} - \frac{2}{3}\rho k \frac{\partial \bar{u}_i}{\partial xj}\delta_{ij} - \beta^* \rho k \omega \tag{2.31}$$

$$\frac{\partial(\rho \omega)}{\partial t} + \frac{\partial}{\partial x_i}(\rho \omega \bar{u}_i) = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_{\omega,1}}\right)\frac{\partial \omega}{\partial x_i}\right] + \gamma_2\left(2\rho S_{ij}S_{ij} - \frac{2}{3}\rho\omega\frac{\partial \bar{u}_i}{\partial xj}\delta_{ij}\right) - \beta_2\rho\omega^2 + \frac{2\rho}{\sigma_{\omega,2}\omega}\frac{\partial k}{\partial x_k}\frac{\partial \omega}{\partial x_k} \tag{2.32}$$

# 3 Computational Fluid Dynamics

CFD is a technique that allows the study of fluid flows and heat transfer phenomena in either simple or complex systems with the help of computer simulations. It has a wide range of applications from industrial to non-industrial or research [9].

Most commercial CFD software packages, including openFOAM used for this thesis, employs the finite volume method (FVM) to represent and evaluate set of partial differential equations that represent the phenomena in the form of algebraic equations. This numerical method first integrates the governing equations over the computational domain control volumes, then the equations are discretized into algebraic equations, to then solve the algebraic equations by means of iteration until convergence is achieved.

## 3.1 openFOAM

As many other CFD software packages, openFOAM consists of three main components:

- Pre-Processing
- Solver
- Post-Processing

In pre-processing the case to be study is specified, this includes the computational domain (grid or mesh), the initial and boundary conditions, and the required thermophysical properties. As well as selection of the solution schemes and controls.

The appropriate solver to be used must be selected based on the case to be study, for this thesis work a solver applicable to transient and compressible fluids flows that involves heat transfer was selected.

After numerical calculation has been completed, the results can be visualized on post-processing. There are several methods to visualize the data, this includes variables fields and plots.

openFOAM software does not have a graphical interface, so the simulation parameters are required to be specified in dictionary files. These files are contained mainly in three separate folders called "0", "constant" and "system" and every case study has the following structure as shown in Figure 3.1.
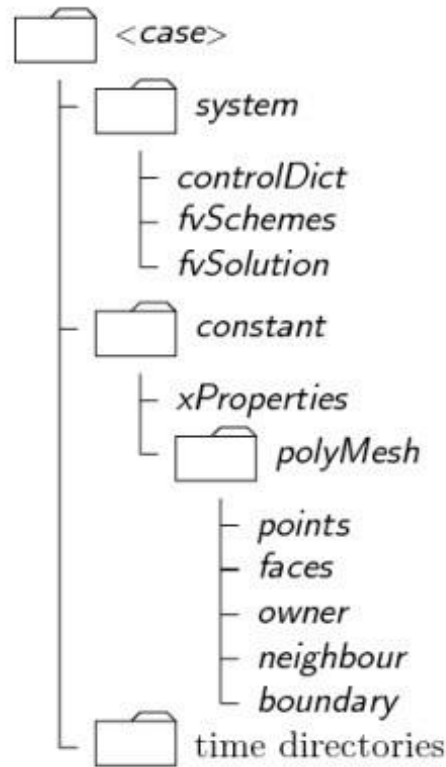
Figure 3.1 Case directory structure

The "0" folder contains the initial and boundary conditions for pressure, temperature, velocity, turbulent kinetic energy, etc. The "constant" folder contains dictionaries that specify the thermophysical and transport properties as well as a subfolder named "polyMesh" which contains information about the geometry domain (mesh). The "system" folder contains all the required parameters for solution control, these include from total simulation time and time step, up to the discretization schemes for all the variables and control solvers for matrices.

## 3.2 Solver

For the work on this thesis the "rhoCentralFoam" solver was selected. As discussed before, this solver is applicable for compressible and transient flows involving heat transfer. The solver was coded based on the work by Kurganov & Tadmor [14]. A short description on the iterative solution algorithm will be presented in this section.

The algorithm starts by calculating the Temperature ($T$), density ($\rho$) and velocity ($u$) from the initial condition to the faces of the control volume by interpolating from the control volume centroids, to then substituting them for the calculation of the convective flux terms.

After, the density ($\rho$) is calculated by solving the continuity equation, as referred in 2.2.

The inviscid momentum velocity is then calculated from the inviscid momentum equation.

$$\left(\frac{\partial \widehat{\boldsymbol{u}}}{\partial t}\right) + div(\boldsymbol{u}\widehat{\boldsymbol{u}}) + div\, p = 0 \qquad (3.1)$$

$$\frac{(\hat{\rho}u) - (\rho u)^n}{\partial t} + div(\boldsymbol{u}\hat{\boldsymbol{u}}) + div\, p = 0 \tag{3.2}$$

Then, the corrected value of velocity ($\hat{u}$) is calculated with the values of $u$ and $\rho$, since $\hat{u} = \rho u$; taking into account the viscous momentum equation:

$$\left(\frac{\partial(\rho\boldsymbol{u})}{\partial t}\right) - div(\mu\, grad\, u) = 0 \tag{3.3}$$

$$\frac{(\rho u)^{n+1} - (\hat{\rho}u)}{\partial t} - div(\mu\, grad\, u) = 0 \tag{3.4}$$

Using the same approach, the energy equation is solved for a predictor value of the energy flux **E** using the inviscid energy equation.

$$\left(\frac{\partial(\rho\boldsymbol{E})}{\partial t}\right) - div[u(\boldsymbol{E} + \boldsymbol{p})] + p\, div\, \boldsymbol{u} = 0 \tag{3.5}$$

The temperature can then be calculated using Equation 3.6, with the values of $\rho$, u and **E**.

$$T = \frac{1}{C_v}\left(\frac{\rho E}{\rho} - \frac{|u^2|}{2}\right) \tag{3.6}$$

And finally, the value estimated for T is used for the corrected energy equation to include the diffusive heat flux.

$$\left(\frac{\partial(\rho C_v T)}{\partial t}\right) - div(k\, grad\, T) = 0 \tag{3.7}$$

The pressure is updated using the real gas equation of state as described in Equations 2.11 to 2.16.

## 3.3  Numerical Schemes

openFOAM offers several numerical schemes to discretize the governing equations in space and time to be numerically solved according to the FVM. In openFOAM one must specify the desired numerical scheme for each of the terms in the governing equations as shown in Figure 3.2:

Figure 3.2 Numerical schemes terms, taken from [15]

The desired schemes must be specified in the *fvScheme* file in the *System* folder. A list of the numerical schemes used for the thesis work are shown in Table 3.1. The selection on the schemes is based on the tutorials included in openFOAM for the rhoCentralFoam solver.

Table 3.1 List of used numerical schemes

| Term | Scheme |
|------|--------|
| ddtSchemes | Euler |
| gradSchemes | Gauss linear |
| divSchemes | div(phi,U)      linearUpwind grad(U)<br>div(phi,k)      Gauss upwind<br>div(phi,omega) Gauss upwind<br>div(tauMC)     Gauss linear |
| laplacianSchemes | Gauss linear corrected |
| InterpolationSchemes | Default linear<br>reconstruct(rho)    vanLeer<br>reconstruct(U)     vanLeerV<br>reconsturct(T)     vanLeer |
| snGradSchemes | corrected |
| wallDist | method meshWave |

## 3.4 Solver control

Similar to the selection of the numerical schemes in the previous subchapter, in openFOAM one must specify the solver for the matrix of algebraic equations. Each iteration generates a residual for a specific variable, and this residual is recalculated for each iteration. Once the residual value is below the specified tolerance, the solver stops. This controls are specified in the *fvSolution* file in the *System* folder. Table 3.2 shows the solution controls used this thesis work.

Table 3.2 List of used solver controls

| Equation | Controls | |
|---|---|---|
| rho, rhoU, rhoE | solver diagonal | |
| k, omega | Solver | smoothSolver |
| | Smoother | symGaussSeidel |
| | Tolerance | 1e-05 |
| | relTol | 0.0 |
| U, h | Solver | smoothSolver |
| | Smoother | symGaussSeidel |
| | nSweeps | 2 |
| | Tolerance | 1e-09 |
| | relTol | 0.0 |

## 3.5  Thermophysical properties

A proper thermodynamic and transport model must be established to ensure that the simulation data represents the real process as possible, as thermophysical properties will vary with temperature. openFOAM offers different thermophysical models depending on the type of fluid being simulated. All the model inputs are located in the *thermophysicalProperties* file, in the *constant* folder. Since pure hydrogen is the focus of this thesis work, the *psiThermo* model was selected as it is developed for a fixed composition based on compressibility [15]. The thermophysical model data inputs can be seen in Table 3.3.

Table 3.3 openFOAM thermophysical model

| Input | Model |
|---|---|
| type | hePsiThermo |
| mixture | pureMixture |
| transport | polynomial |
| thermo | hPolynomial |
| equationOfState | PengRobinsonGas |
| energy | sensibleEnthalpy |

The coefficients for the thermophysical model in the form of polynomial equations 3.8, 3.9 and 3.10 for the heat capacity, viscosity and thermal conductivity were extracted from literature [16, 17] and are shown in Table 3.4, Table 3.5 and Table 3.6, for different range of temperature (RoT) each.

$$Cp(T) = A + BT + CT^2 + DT^3 \tag{3.8}$$

$$\mu(T) = A + BT + CT^2 + DT^3 + ET^4 + FT^5 + GT^6 \tag{3.9}$$

$$\kappa(T) = A + BT + CT^2 + DT^3 + ET^4 \qquad (3.10)$$

Table 3.4 Specific heat (Cp) polynomial coefficients.

| RoT [K] | A | B | C | D ($10^{-4}$) | E | F | G |
|---|---|---|---|---|---|---|---|
| 100 - 365 | 6436.5105 | 63.161307 | -0.1685728 | 1.5229265 | 0 | 0 | 0 |

Table 3.5 Dynamic viscosity (μ) polynomial coefficients

| RoT [K] | A | B ($10^{-2}$) | C ($10^{-4}$) | D ($10^{-6}$) | E ($10^{-9}$) | F ($10^{-12}$) | G ($10^{-15}$) |
|---|---|---|---|---|---|---|---|
| 10 - 500 | -0.135666 | 6.84115878 | -3.928747 | 1.8996 | -5.23104 | 7.4490972 | -4.250937 |

Table 3.6  Thermal conductivity (κ) polynomial coefficients

| RoT [K] | A ($10^{-2}$) | B ($10^{-4}$) | C ($10^{-6}$) | D ($10^{-9}$) | E ($10^{-12}$) | F | G |
|---|---|---|---|---|---|---|---|
| 100 - 500 | 2.009705 | 3.234622 | 2.1637249 | -6.49151204 | 5.52407932 | 0 | 0 |

For the Peng-Robinson EOS, the data inputs used are shown in Table 3.7 [18], where $T_c$, $P_c$ and $V_c$ are the critical temperature, pressure and volume, respectively. And $\omega$ is the acentric factor.

Table 3.7 EOS parameters

| EOS | Tc [K] | Pc [MPa] | Vc [m3/kmol] | ω |
|---|---|---|---|---|
| Peng-Robinson | 32.994 | 1293920.25 | 65.5e-3 | -0.220 |

# 4 Modelling set-up

## 4.1 Simulation cases

A set of different cases with different inlet mass flow rate were defined. Due to a low mass flow coupled with a large tank volume, the total simulation time for a complete tank refueling operation would require a lot of computational time. Because of this, each of the cases were set up for a total time of 20 seconds with different initial pressure values ranging from 1 MPa to 35 MPa. The set of cases for the intervals of initial pressure were grouped according to the inlet mass flow rate. This approach has the drawback that not all the history of the process will be captured between the pressure intervals used. Nevertheless, the goal was to analyze the temperature and pressure distribution, as well as the heat transfer coefficient between the gas and wall. The simulations cases performed are shown in Table 4.1 and Table 4.2.

Table 4.1 Group 1 – 10 g/s mass flow

| Case | Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|------|----------------------|----------------------|------------------------|
| 1 | 273.15 | 10 | 1 |
| 2 | 273.15 | 10 | 5 |
| 3 | 273.15 | 10 | 10 |
| 4 | 273.15 | 10 | 20 |
| 5 | 273.15 | 10 | 30 |
| 6 | 273.15 | 10 | 35 |

Table 4.2 Group 2 – 50 g/s mass flow

| Case | Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|------|----------------------|----------------------|------------------------|
| 7 | 273.15 | 50 | 1 |
| 8 | 273.15 | 50 | 5 |
| 9 | 273.15 | 50 | 10 |
| 10 | 273.15 | 50 | 20 |
| 11 | 273.15 | 50 | 30 |
| 12 | 273.15 | 50 | 35 |

## 4.2 Mesh

The mesh was created using the included openFOAM blockMesh utility. The overall computational domain is composed by several three-dimensional hexahedral blocks. First the vertices are defined. Second, the hexahedral blocks are defined from the vertices and the desired number of cells in each block. Third, the boundary patches and walls are defined on the blocks' surfaces. Since openFOAM is not able to use real 2-D meshes, the set up for performing a 2-D simulation consists in defining a unitary cell size on the Z direction. In that way, the geometry is three dimensional but with a single layer thickness. Considering the 2-D representation of a cylinder, the computational domain has a rectangular geometry as shown in Figure 4.1.



Figure 4.1 Vertices (points) and blocks of the computational domain

The dimensions of the simulated tank are shown in Figure 4.2. The walls of the mesh are divided into three regions:

- Inlet region
- Middle region
- Far region
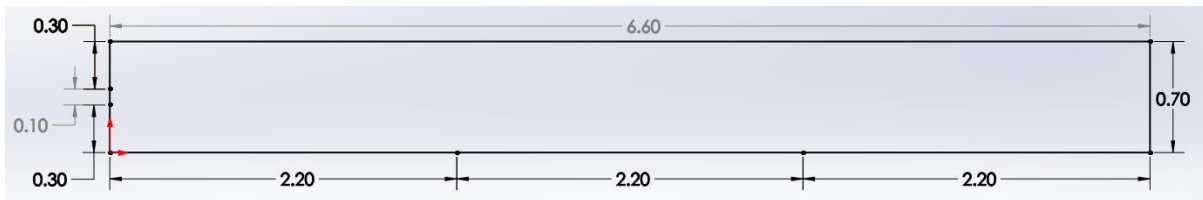


Figure 4.2 Domain dimensions

Each wall region has a length of 2.2 meters along the tank center line axis. The goal is to better visualize the heat wall transfer properties in the regions near the inlet, middle and far away from it. A summary of the dimensions is shown in Table 4.3.

Table 4.3  Mesh dimensions

| Feature | Dimension (m) |
|---|---|
| Tank diameter | 0.7 |
| Tank length | 6.6 |
| Inlet diameter | 0.1 |
| Inlet region length<br>Middle region length<br>Far region length | 2.2 |

The total number of two-dimensional cells is 31500. This number provides sufficient resolution without a huge increase in computational time. It was found on previous simulations that a coarser mesh with a smaller number of cells creates large oscillations after a few seconds of simulated time. A finer mesh creates generates smaller oscillations but with an increased computational time as a drawback. Close up sections of the mesh are shown in Figure 4.3 and Figure 4.4.
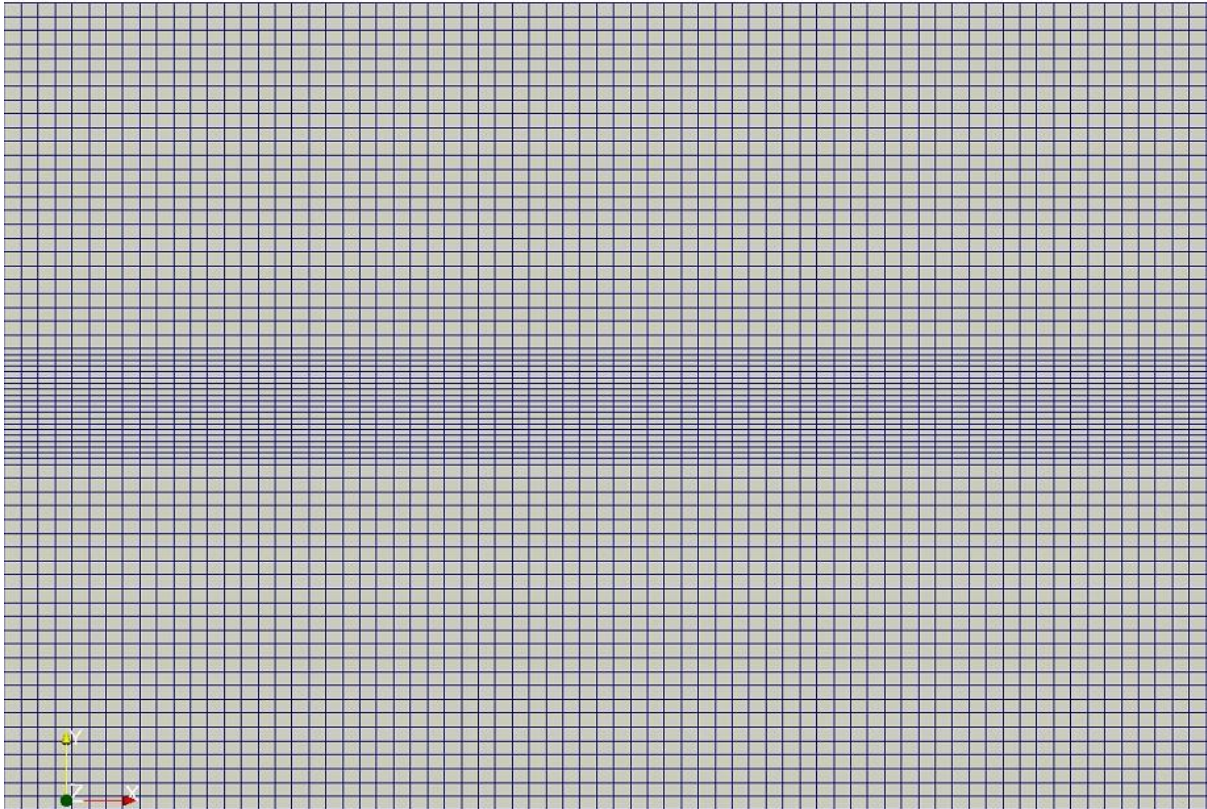


Figure 4.3 Hexahedral mesh
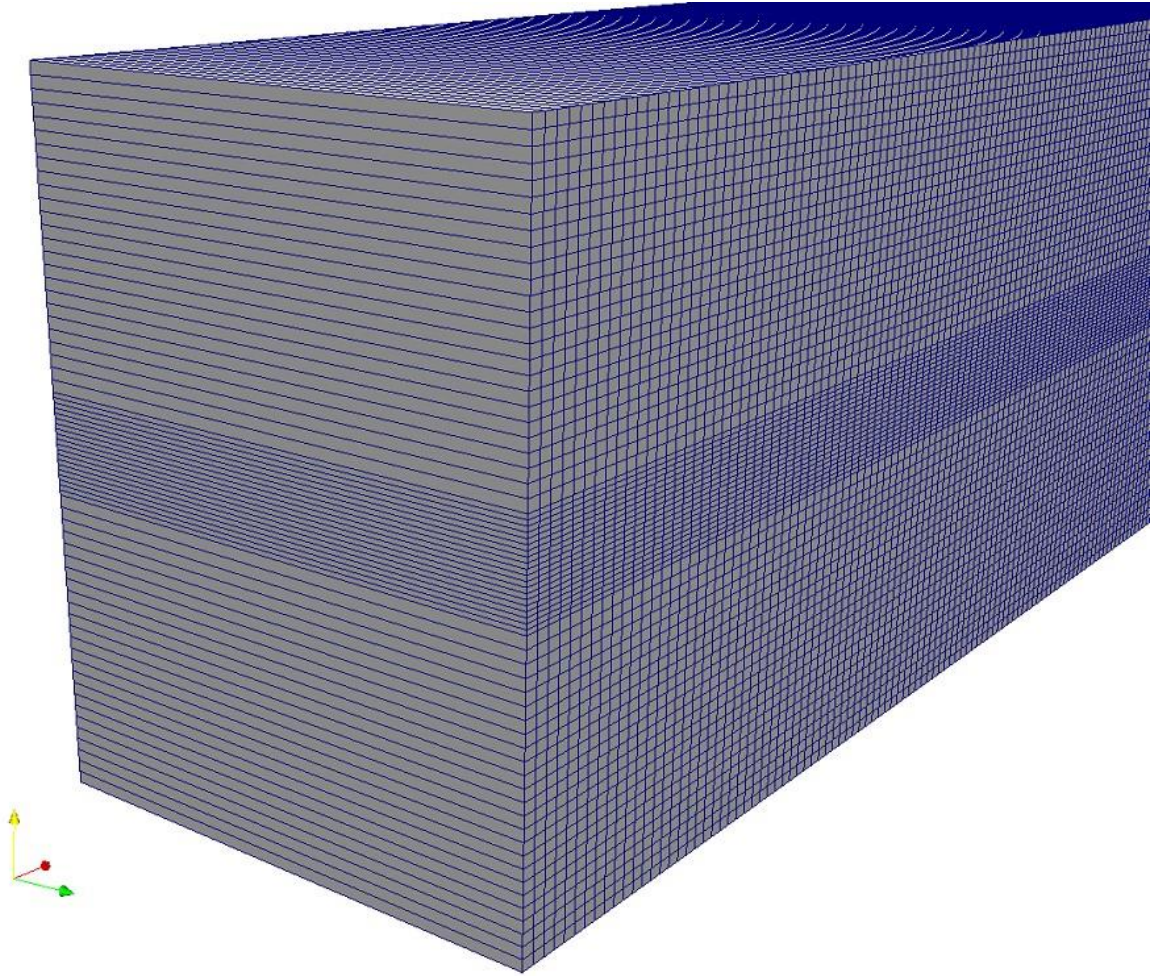
Figure 4.4 Single layer mesh

## 4.3  Boundary & Initial Conditions

The boundary conditions (BC) must be specified for the computational mesh. These BCs can be of several type depending on the simulation to be performed. As previously discussed, the walls of the simulated tank were decomposed into three regions. This regions were assigned a *Wall* type boundary condition, as fluid do not cross those boundaries. While the inlet was assigned as a *Patch* boundary condition, allowing fluid to cross. Figure 4.5 shows the inlet patch BC in red as well as the top three wall regions in green, blue, and yellow.
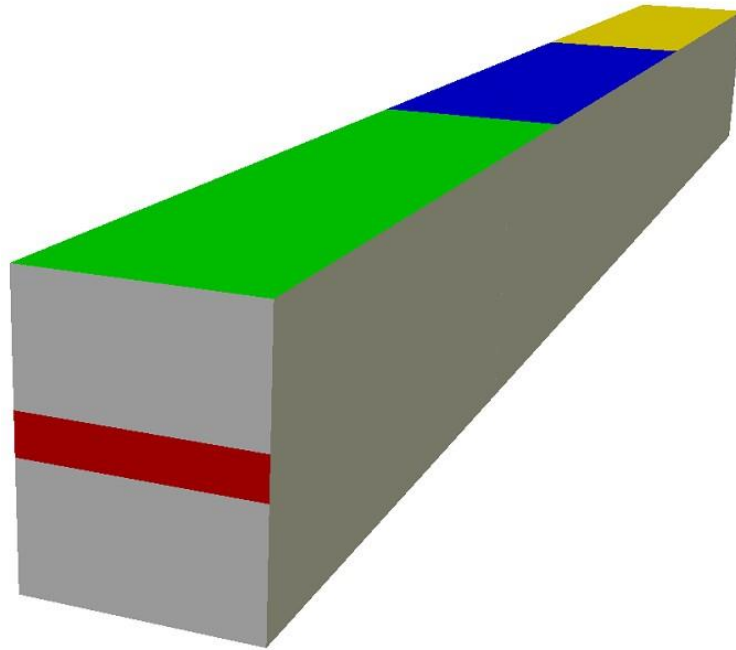
Figure 4.5 Boundary patches: inlet (red), inlet wall (green), middle wall (blue) and far wall (yellow)

Table 4.4 shows the summary of the boundary conditions used for all the twelve (12) cases.

Table 4.4 Boundary conditions type

| Patch | p | U | T |
|---|---|---|---|
| inlet | zeroGradient | flowRateInletVelocity | fixedValue |
| inletupperWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| middleupperWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| farupperWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| inletlowerWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| middlelowerWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| farlowerWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| rightWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| leftWall | zeroGradient | noSlip | externalWallHeatFluxTemperature |
| defaultFaces | empty | empty | empty |

As previously discussed, the k-ω SST turbulence model introduces two new field variables. Based on equations 2.28 and 2.30 it is possible to approximate the boundary conditions for the turbulent kinetic energy $k$ and the specific rate of dissipation $\omega$.

For the turbulent kinetic energy $k,$ a value of mean inlet flow velocity is needed for each of the different simulations performed. Figure 4.6 shows the velocity at the inlet patch through the simulation time for case 7. Velocity decreases to compensate for the increase in pressure and density at the inlet. The highest value of velocity in the profile for each case was chose as the mean velocity for the turbulent boundary conditions calculation.
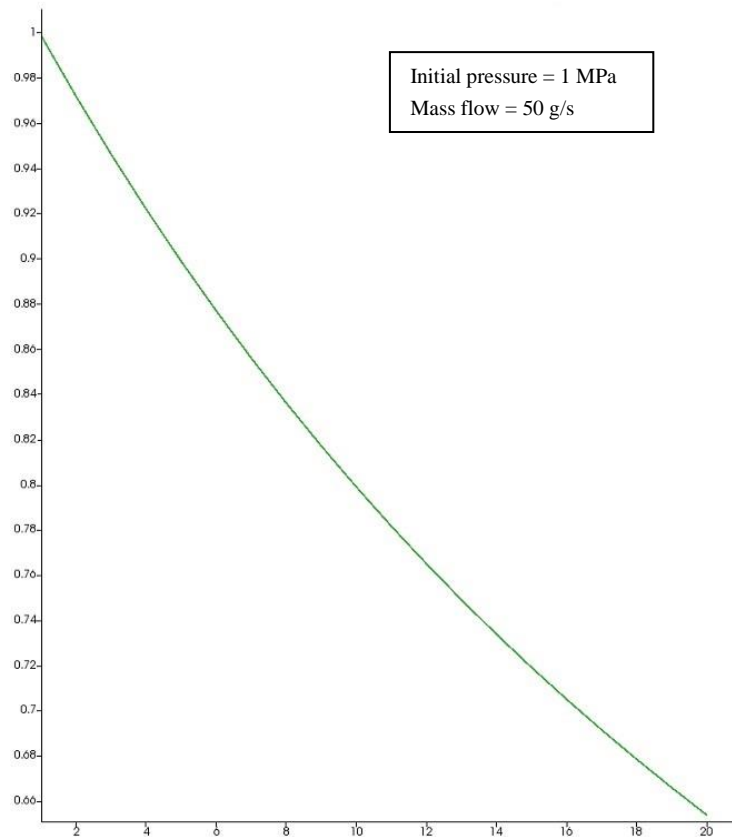
Figure 4.6 Velocity profile (m/s) with time (s) at inlet

Table 4.5 shows the boundary conditions used for the turbulence parameters. The wall function values defined are only used as educated guesses. Hydrogen properties were obtain from [19], [20] and [21].

Table 4.5 Boundary conditions for turbulence

| Patch | k | ω |
|---|---|---|
| internalField | uniform 0.0001 | uniform 0.0001 |
| inlet | fixedValue<br>value uniform | fixedValue<br>value uniform |
| inletupperWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| middleupperWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| farupperWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| inletlowerWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| middlelowerWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| farlowerWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| rightWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| leftWall | kqRWallFunction;<br>value uniform 0.0001 | omegaWallFunction;<br>value uniform 0.0001 |
| defaultFaces | empty | empty |

For the thermal boundary conditions, the externalWallHeatFluxTemperature condition was used, as it can be used to specify a convective heat transfer boundary condition by defining the external heat transfer coefficient and the ambient temperature. Taking into account the total thermal resistance of convective and conductive heat transfer. For this the following parameters were defined as shown in Table 4.6.

Table 4.6 Thermal boundary layer parameters

| Ambient Temperature [K] | Wall thickness [m] | Wall thermal conductivity [W/m k] | Ambient heat transfer coefficient [W/m$^2$ K] |
|---|---|---|---|
| 298.15 | 0.005 | 0.4 | 10 |

# 5 Results and discussion

This chapter summarizes the most significant simulation findings, and sources of uncertainty are identified. Appendix A contains all the results for each case individually.

## 5.1 Temperature and pressure

The average tank temperature and pressure profile over the whole simulated time were compared and shown in Figure 5.1 to Figure 5.6. Each graph shows a comparison of just two different cases. The cases compared correspond to the same initial pressure condition but with different mass flow rate inlet, 0.01 kg/s vs 0.05 kg/s. As shown in Table 4.1 and Table 4.2.
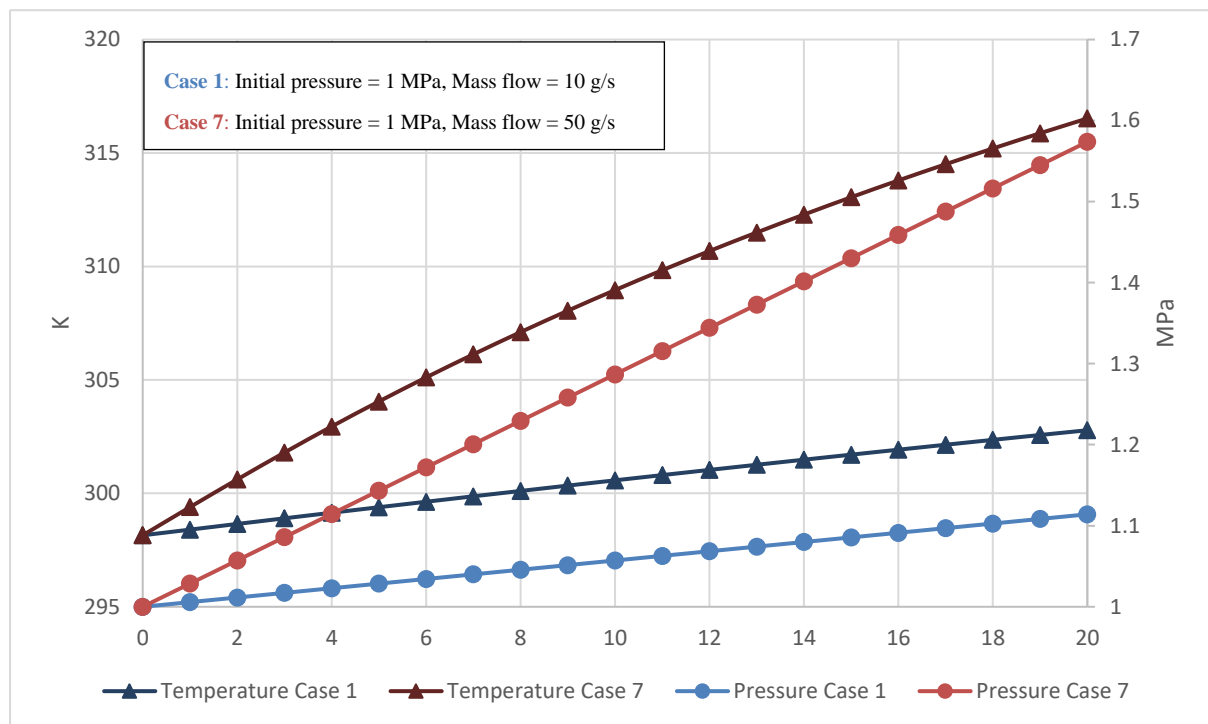


Figure 5.1 Temperature and pression comparison between case 1 and 7

The temperature and pressure trend shown in Figure 5.1 shows an expected increase due to the gas compression inside the tank. Due to the lower mass flow rate at the inlet (0.01 kg/s) of case 1 as compared to case 7 (0.05 kg/s), the final temperature and pressure reached is lower in case 1, with an increase of 4.6 K and 0.11 MPa, respectively. On the other hand, for case 7 the increase is 18 K and 0.57 MPa, respectively. It is expected that the temperature and pressure will continue to increase steadily if the simulation time is extended, but due to time constraints this was not possible.
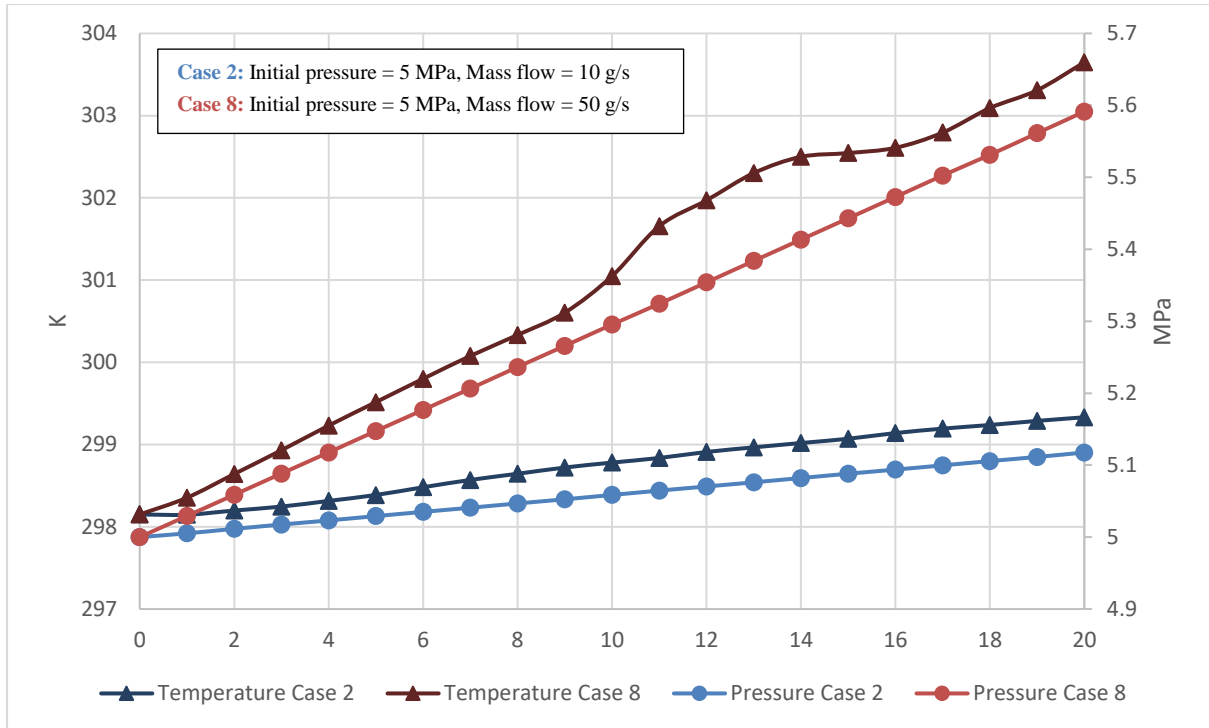
Figure 5.2 Temperature and pression comparison between case 2 and 8

Figure 5.2 shows a similar trend for the temperature and pressure increase inside the tank, now with an initial pressure of 5 MPa. Comparing cases 2 and 8, it is shown that a lower mass flow rate of 0.01 kg/s leads to an increase of 1.2 K and 0.11 MPa, respectively. While case 8 with a mass flow rate of 0.05 kg/s has an increase of 5.5 K and 0.6 MPa, respectively.
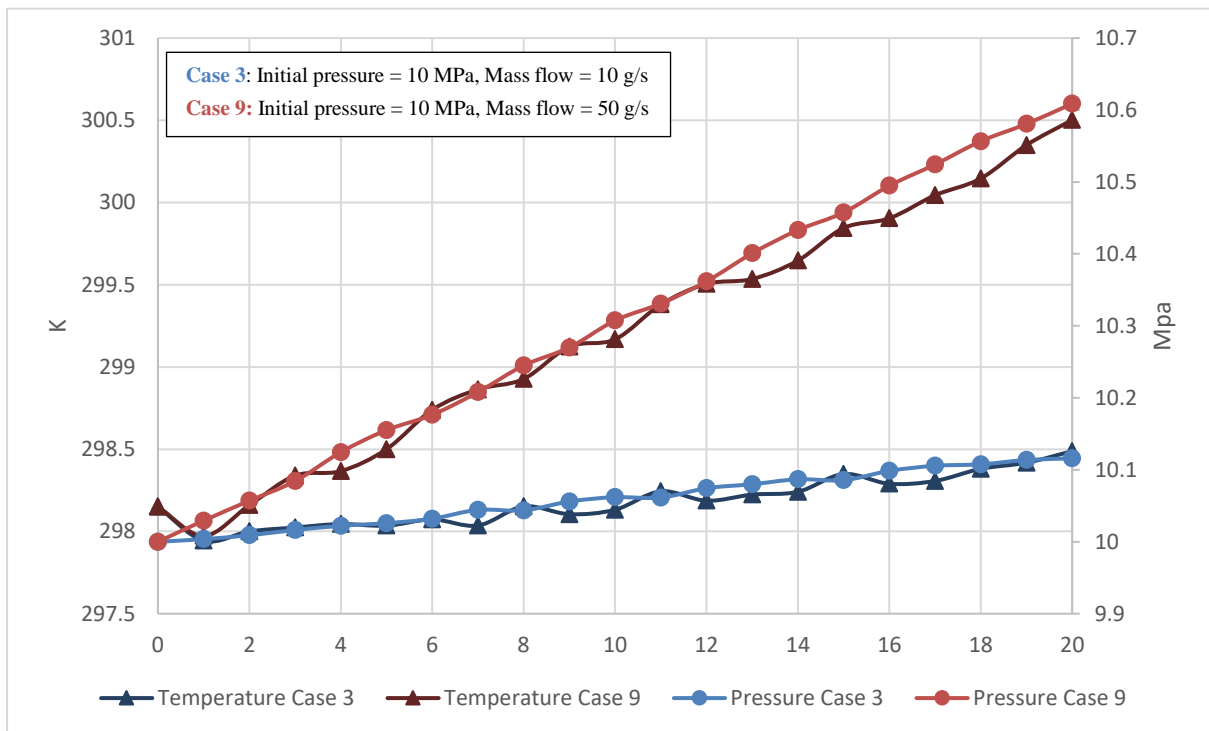


Figure 5.3 Temperature and pression comparison between case 3 and 9
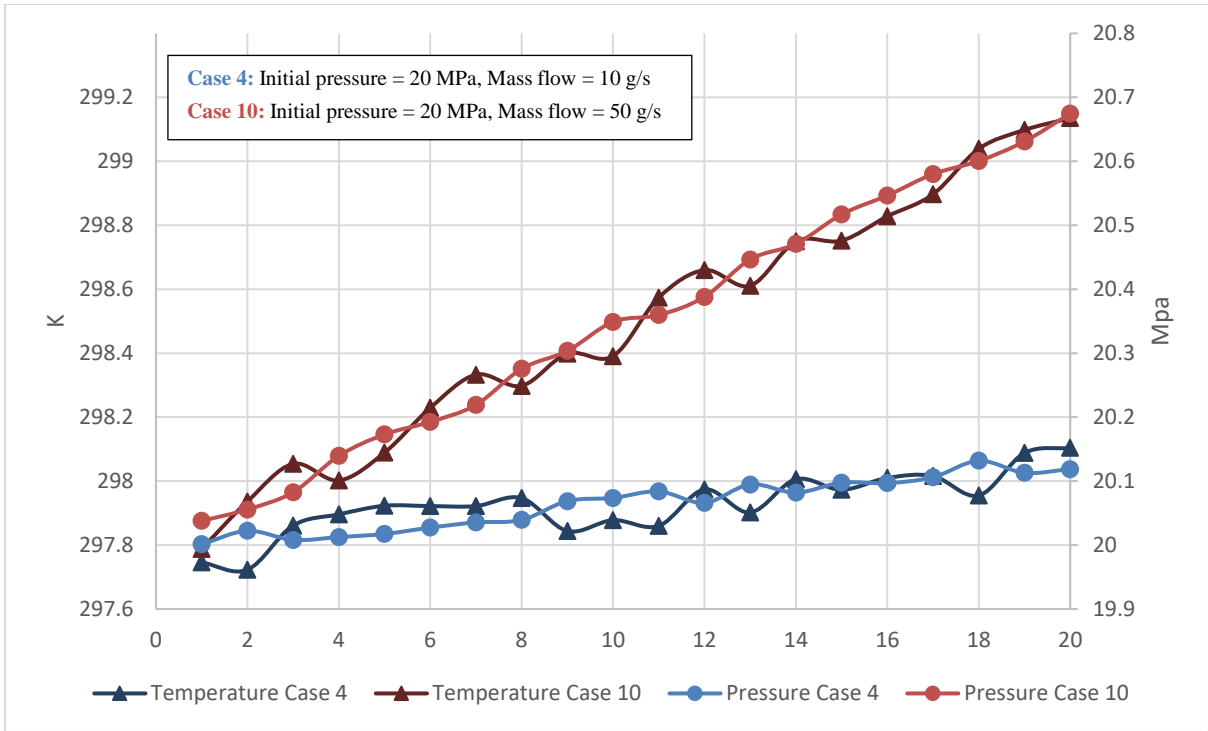
Figure 5.4 Temperature and pression comparison between case 4 and 10
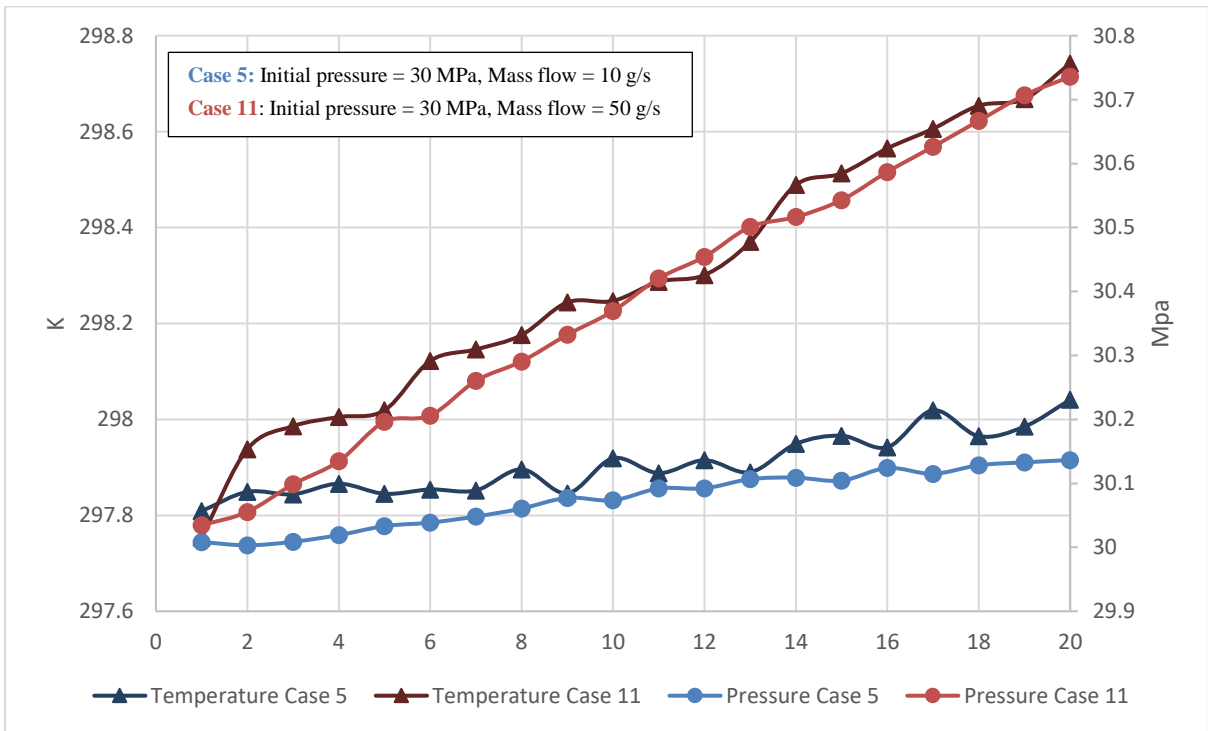


Figure 5.5 Temperature and pression comparison between case 5 and 11
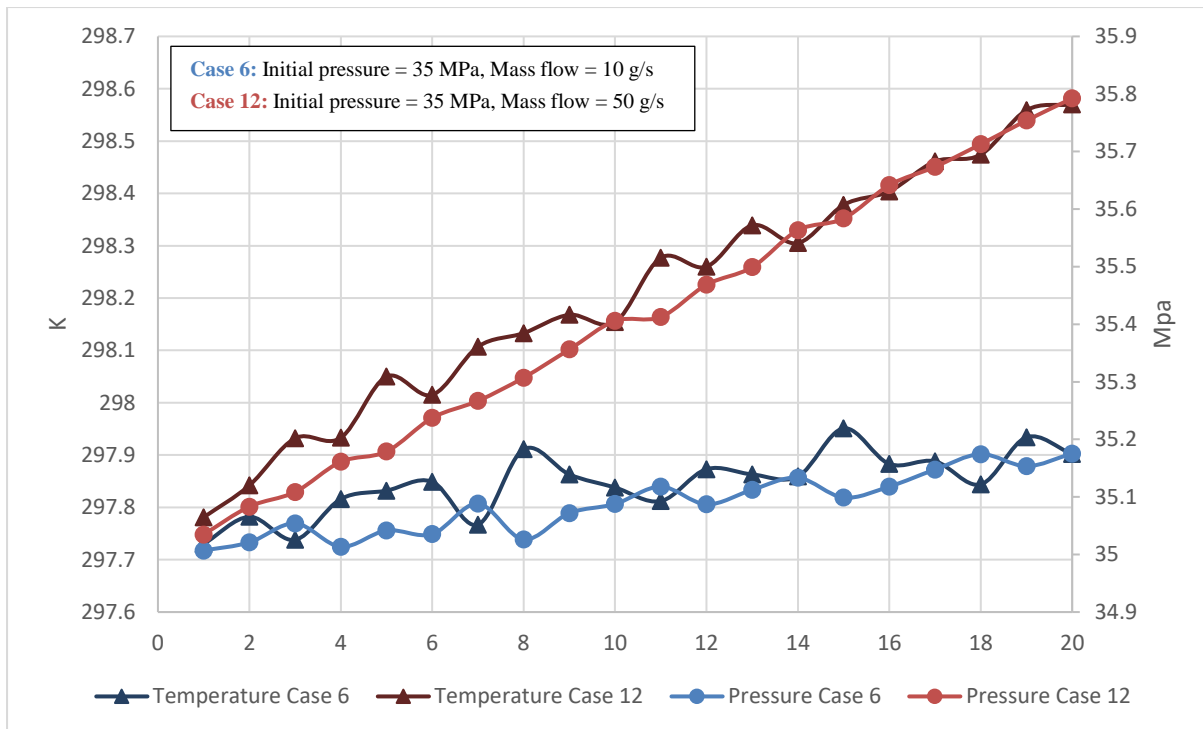
Figure 5.6 Temperature and pression comparison between case 6 and 12

Figure 5.3 to Figure 5.6 show an oscillating trend but an overall increase of temperature and pressure inside the tank, almost neglectable. This oscillating behavior its attributed to cooler regions developed in the middle region within the tank. This cooler regions are only developed for cases with higher initial pressure, literature research does support the development of this cooler regions in the center of the tank when the inlet is located inside the tank, not near the walls as this work. Mesh size was also seen to affect the temperature and pressure distribution within the tank for higher initial pressure cases. These regions eventually heat up due to gas compression and the average temperature increases.

## 5.2 Temperature change vs. initial pressure

The results suggest that with higher values of initial pressure inside the tank, the instant rise on temperature for the first seconds of filling is lower. The temperature change for all cases can be seen in Figure 5.7 and Table 5.1. This inversely proportional behavior it is agreement with the findings of Hosseini et al [3] and Heitsch et al [4] and , in which they found a maximum temperature change at 0.5 MPa initial pressure and the lowest at 2 MPa.
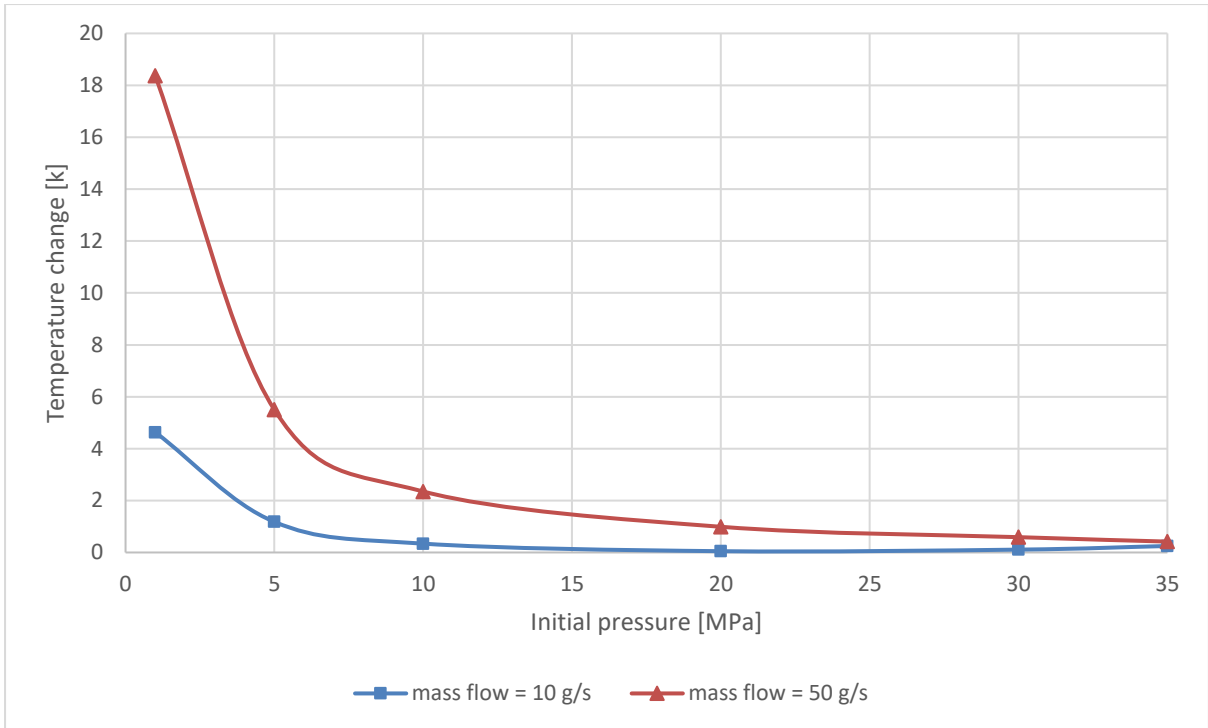
Figure 5.7 Total rise in temperature for different initial pressures
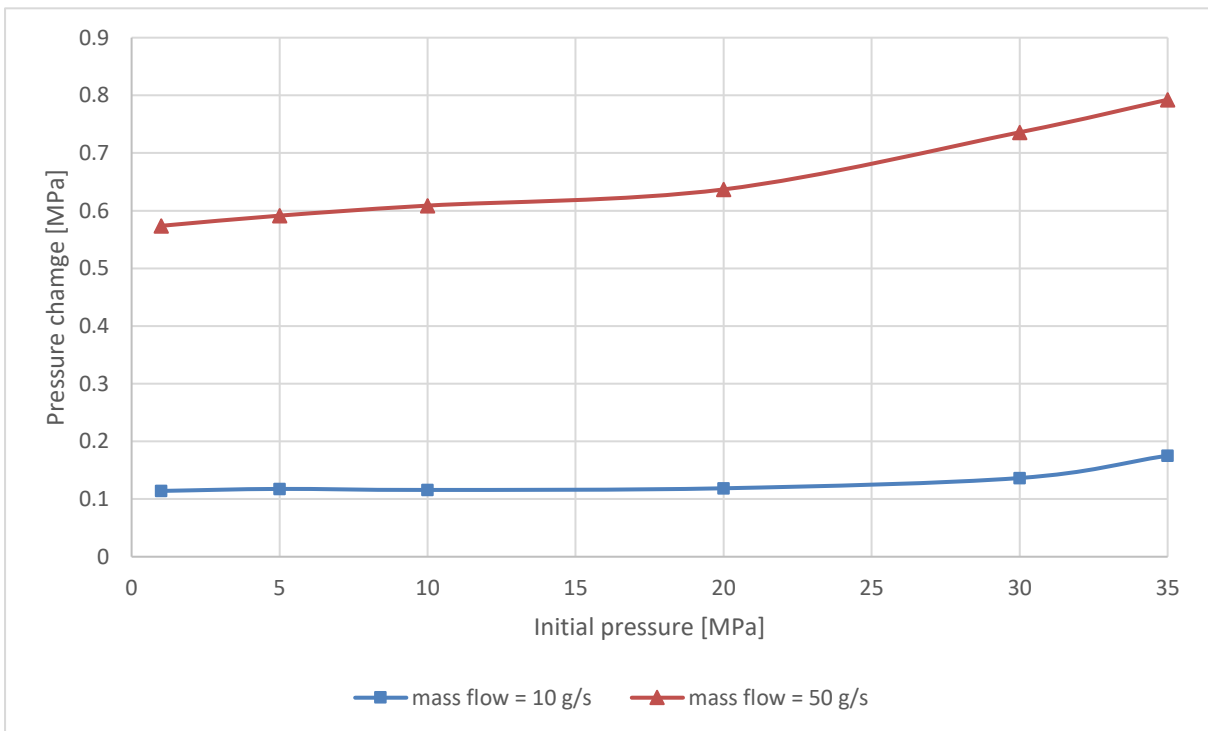


Figure 5.8 Total change in pressure for different initial pressures

Table 5.1 Temperature change

| Case | Initial Pressure [MPa] | Initial Temperature [K] | Final Temperature [K] | Change in Temperature [K] |
|------|------------------------|-------------------------|------------------------|---------------------------|
| 1 | 1 | 298.15 | 302.78 | 4.63 |
| 2 | 5 | 298.15 | 299.33 | 1.18 |
| 3 | 10 | 298.15 | 298.49 | 0.34 |
| 4* | 20 | 298.15 | 298.10 | 0.05 |
| 5* | 30 | 298.15 | 298.04 | 0.11 |
| 6* | 35 | 298.15 | 297.90 | 0.25 |
| 7 | 1 | 298.15 | 316.52 | 18.37 |
| 8 | 5 | 298.15 | 303.65 | 5.50 |
| 9 | 10 | 298.15 | 300.50 | 2.35 |
| 10 | 20 | 298.15 | 299.14 | 0.99 |
| 11 | 30 | 298.15 | 298.74 | 0.59 |
| 12 | 35 | 298.15 | 298.57 | 0.42 |

The marked cases on Table 5.1 showed a lower, almost neglectable final temperature value compared to the initial temperature. This is attributed to the higher initial pressure in those simulations and the mixing of the cooler inlet gas temperature. On the other hand, the cases with the same initial pressure as the marked ones, but with a higher mass flow rate, showed a slightly higher final temperature. Still, the temperature increase is almost neglectable. For all cases it is expected that the temperature inside the tank is increase with longer simulation times.

## 5.2.1 Literature comparison
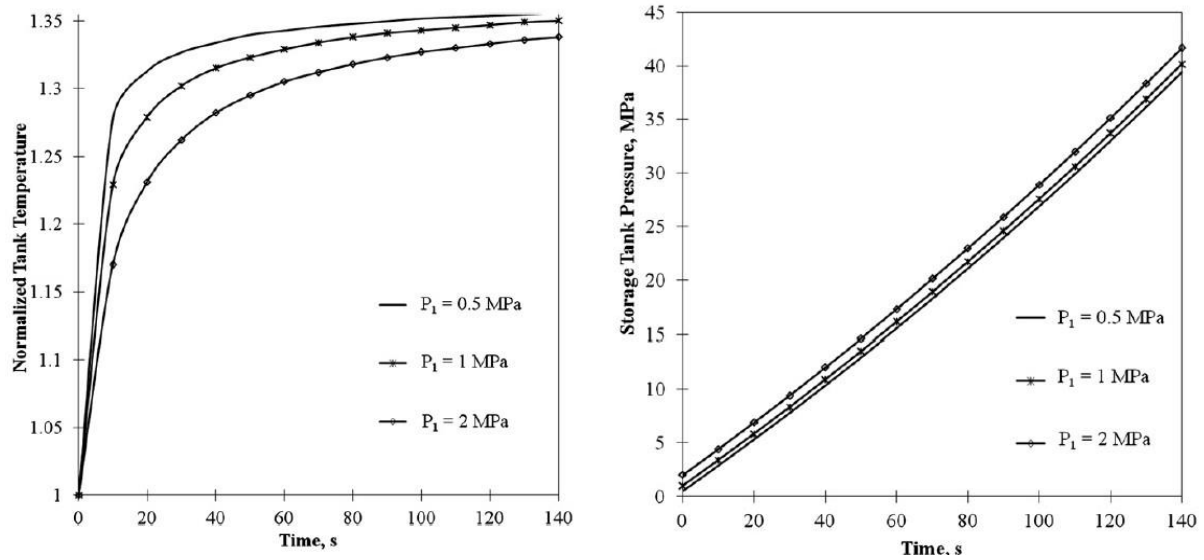


Figure 5.9 Tank temperature change and pressure with time [3]
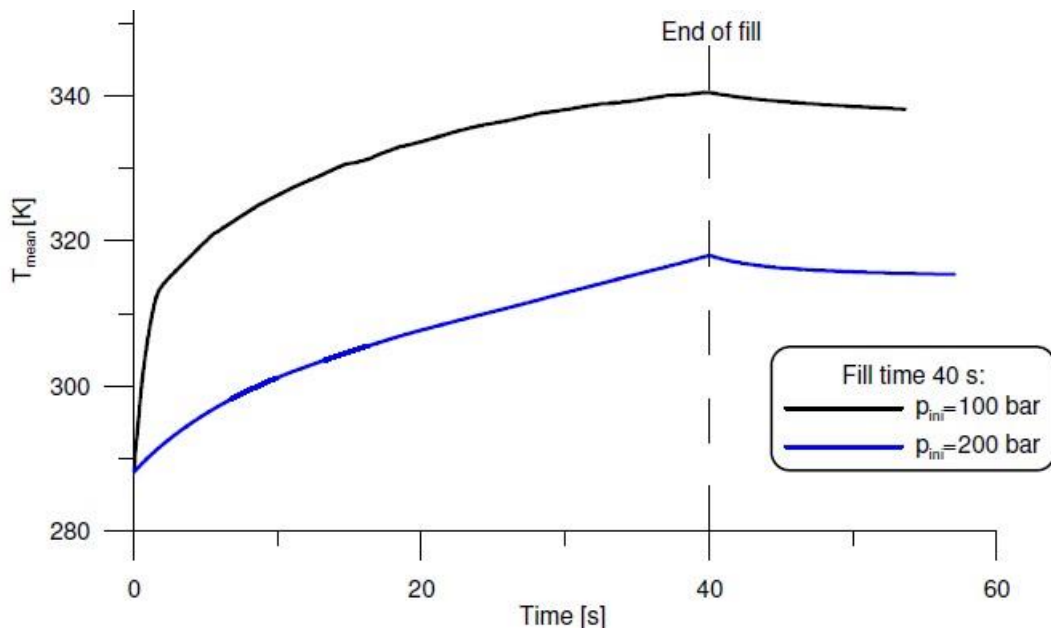


Figure 5.10 Mean temperature vs. initial pressure [4]

Figure 5.9 and Figure 5.10 show the results from Hosseini et al [3] and Heitsch et al [4] for the tank temperature and pressure change with time at different initial pressures. They show an agreement with the results obtained in this work, as higher initial pressures lead to lower instant change in tank temperature, and higher changes in tank pressure. Quantitatively, results differ from this work as the referenced research papers were performed on a smaller tank with different mass flow and for longer times.

Figure 5.11 and Figure 5.12 show the overall temperature field across the tank for case 1 and case 7. For both cases, a concentration of hot gas due to compression is developed at the end of the tank, in the far region opposite to the inlet. This results in a greater temperature located in that far end region. With higher values with greater gas mass flow.
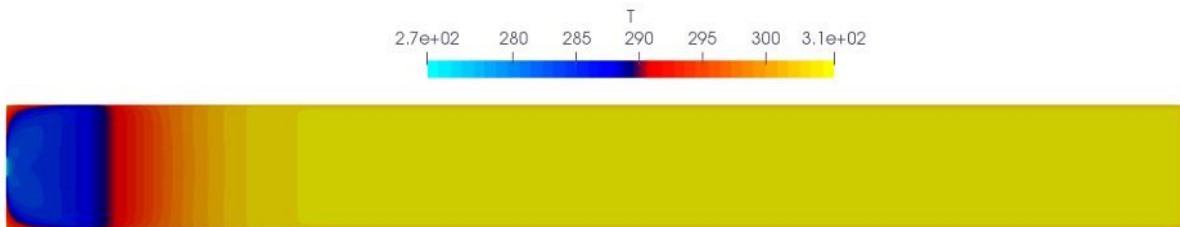


Figure 5.11 Case 1 (1 MPa initial pressure and 10 g/s mass flow) tank temperature flow field at last time step.
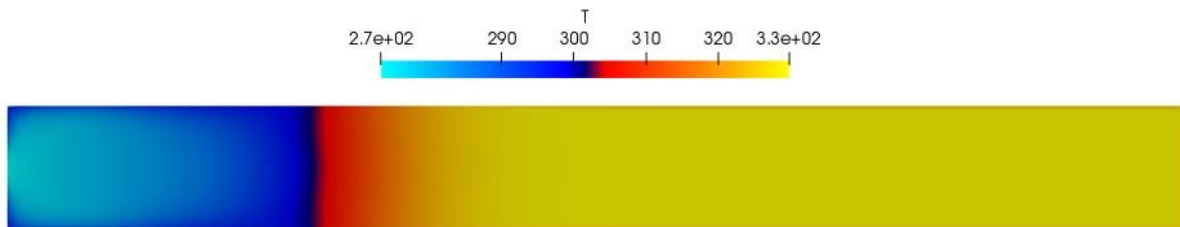


Figure 5.12 Case 7 (1 MPa initial pressure and 50 g/s mass flow) tank temperature flow field at last time step.

## 5.3 Heat transfer coefficient

The heat flux and the heat transfer coefficient at the walls were calculated for all the cases. For the heat flux, the openFOAM utility *"wallHeatFlux"* was used, as it can record the heat flux at every desired patch. The sign convention for the utility is a negative flux for heat going out of the system and positive flux for heat coming into the system. For making the graphs the negative values of the heat flux where change to positive values and vice versa. This means, in this thesis report a positive heat flux means heat going out of the system. With the known heat flux field at the walls, the heat transfer coefficient was calculated using equation 2.19 and a reference temperature.

The reference temperature for thesis report is obtained as the average of the temperatures at the middle and far region for the simulated time and is constant for the coefficient calculations. These temperatures were obtained with the aid of probes.

Appendix A contain all the data and results from all cases.
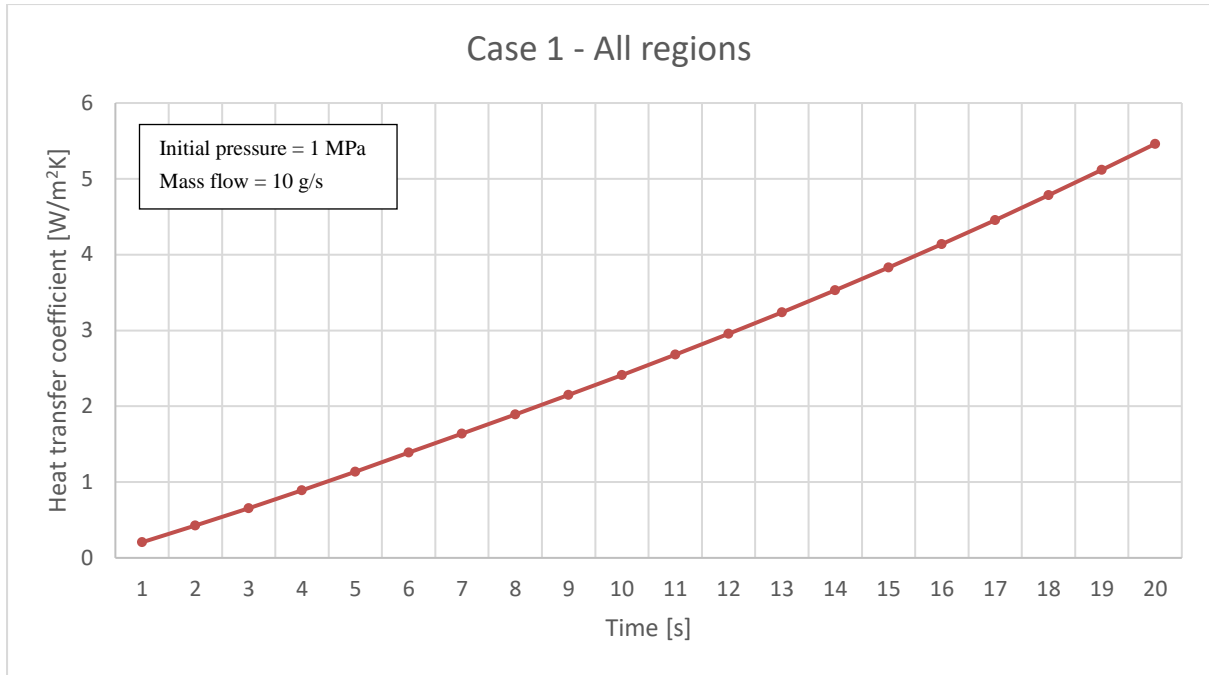
Figure 5.13 Heat transfer coefficient for case 1

Figure 5.13 shows the overall heat transfer coefficient for case 1, for all the tank walls. It has a steady increase for all the simulation time; as the initial condition for the walls is ambient temperature (298.15 K), the difference between the reference temperature and the walls temperature is larger during the initial seconds of the filling. As the gas compresses and heats up, this heat exchange driving force of delta of temperatures reduces leading to an increase in the heat transfer coefficient.
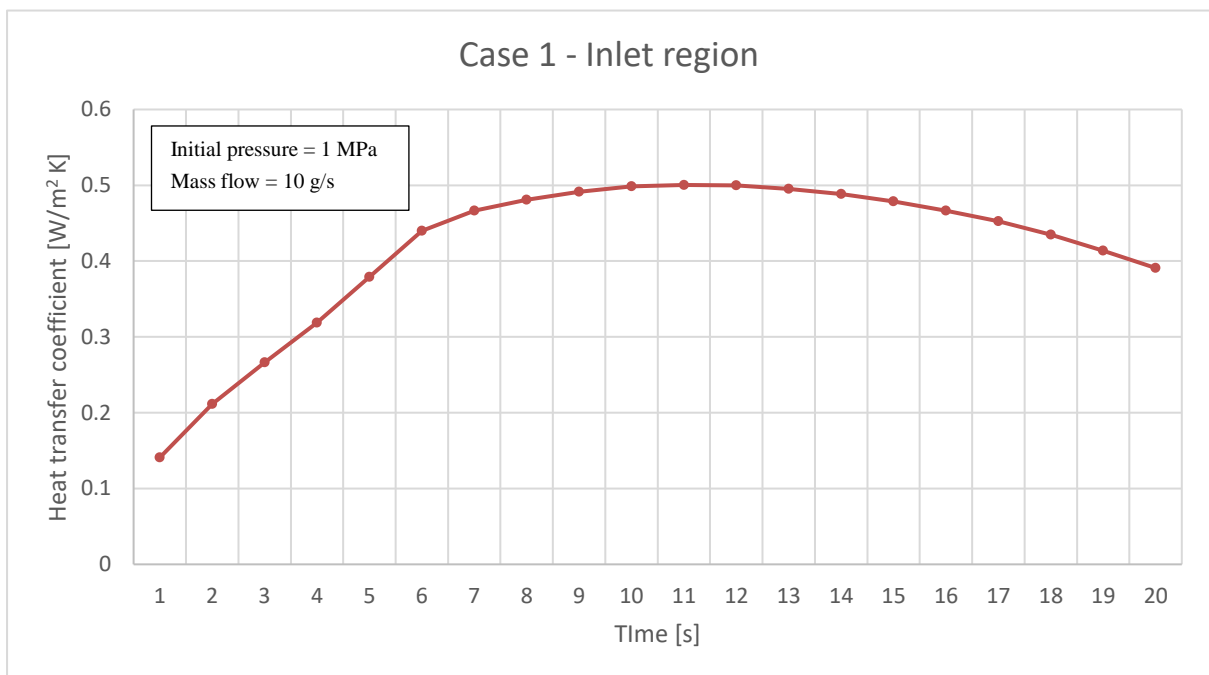


Figure 5.14 Heat transfer coefficient for case 1 at inlet region
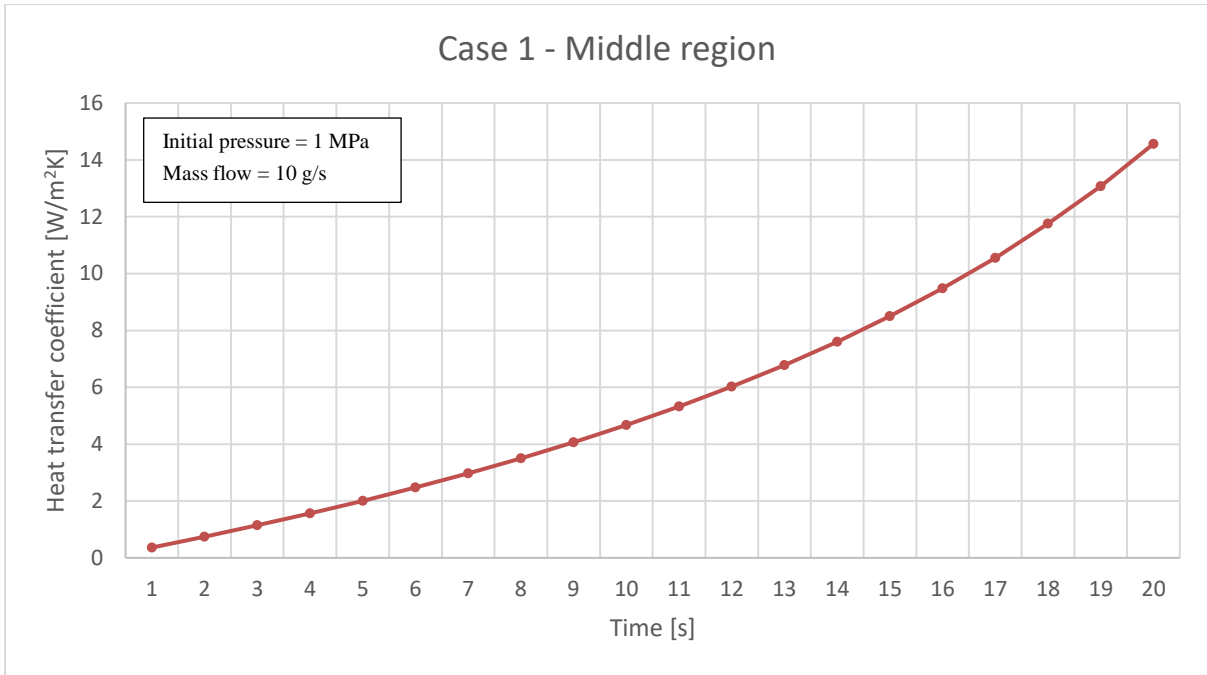
Figure 5.15 Heat transfer coefficient for case 1 at middle region
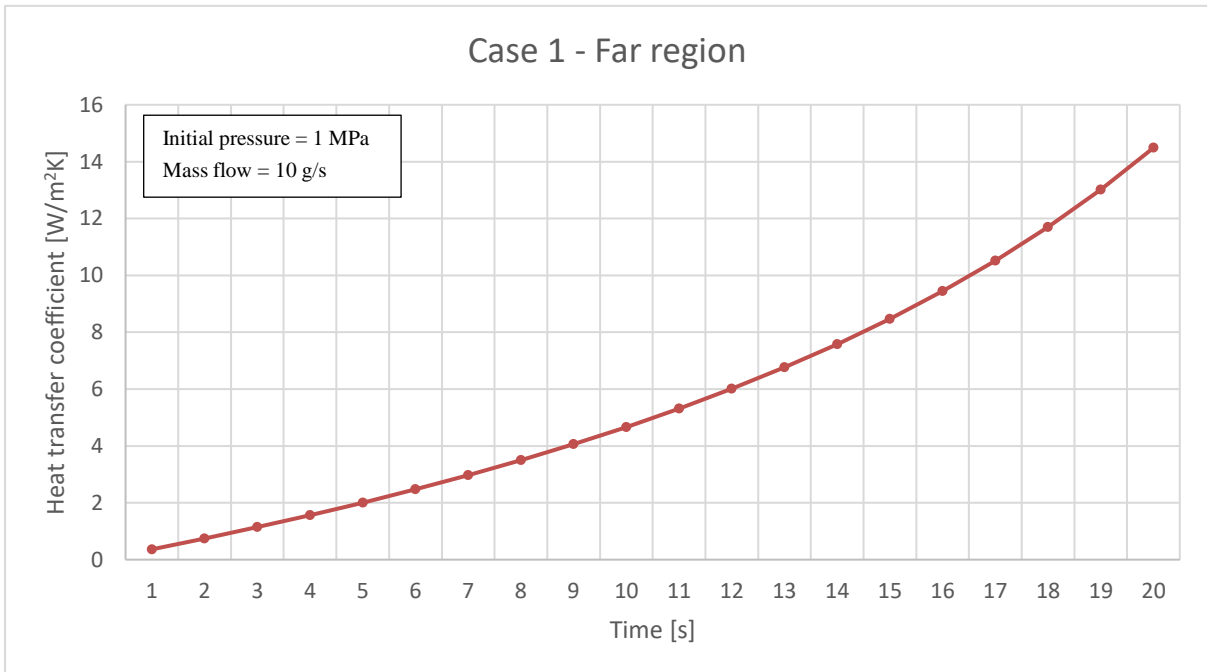


Figure 5.16 Heat transfer coefficient for case 1 at far region

Similarly, Figure 5.14, Figure 5.15 and Figure 5.16 show the heat transfer coefficient for the tank walls at the inlet, middle and far region, respectively. Results at the inlet region showed a steady increase of the heat transfer coefficient but achieving much lower values as compared to the middle and far region. This is attributed to a decrease of heat flux at the inlet region due to inflow of gas at 273.15 K, creating a lower temperature region.

Middle and far wall regions showed almost identical behavior for the heat transfer coefficient. This is in agreement with the fact that inflow gas is compressed at the far region and then in the middle region during the first seconds of filling.
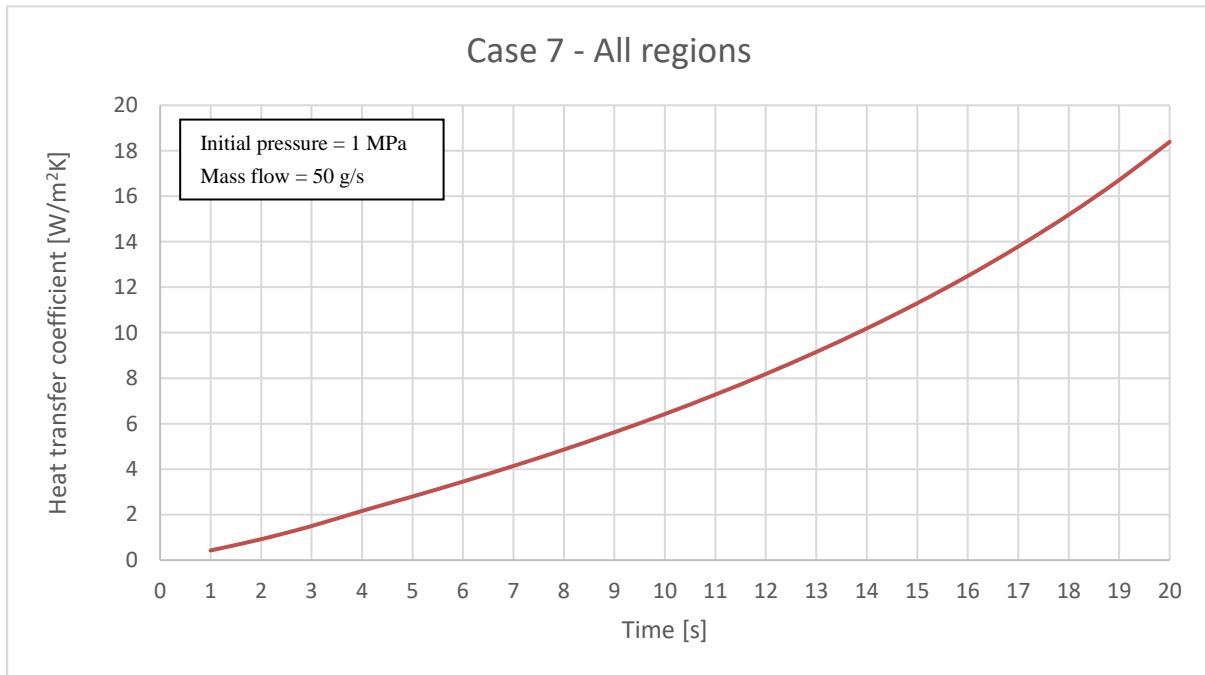


Figure 5.17 Heat transfer coefficient for case 7

For case 7, Figure 5.17 shows the overall heat transfer coefficient for all the walls. A higher gas mass flow of 50 g/s leads to a higher overall convective heat transfer coefficient within the tank.
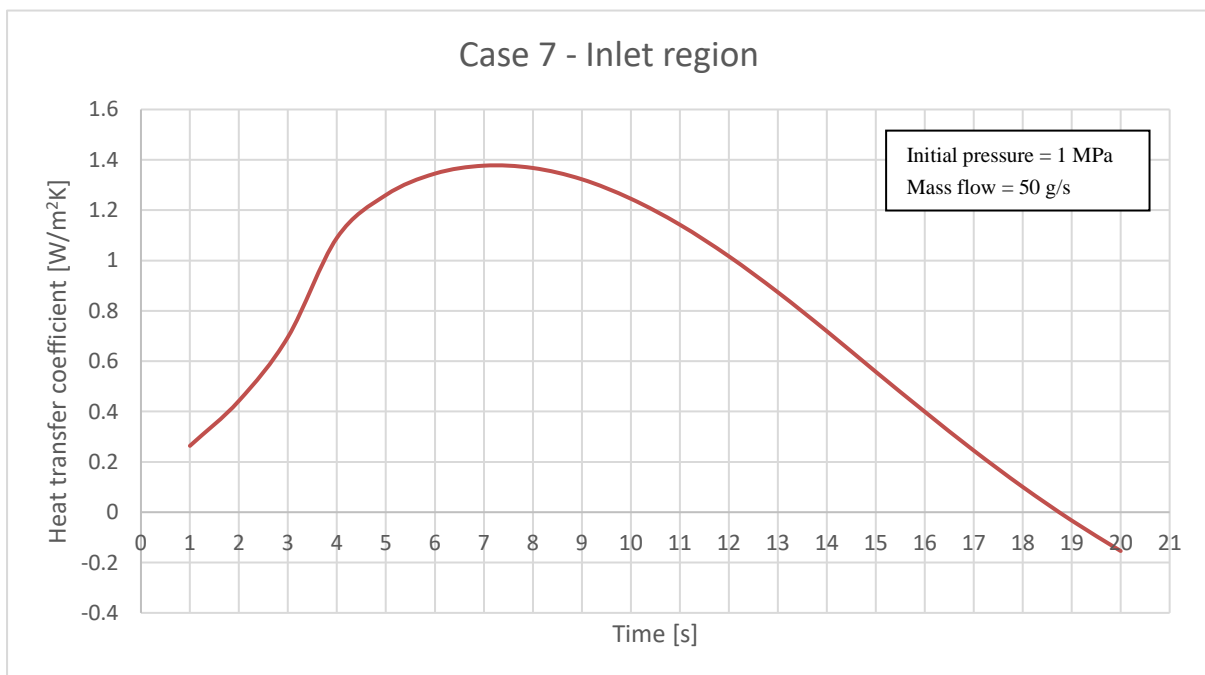


Figure 5.18 Heat transfer coefficient for case 7 at inlet region
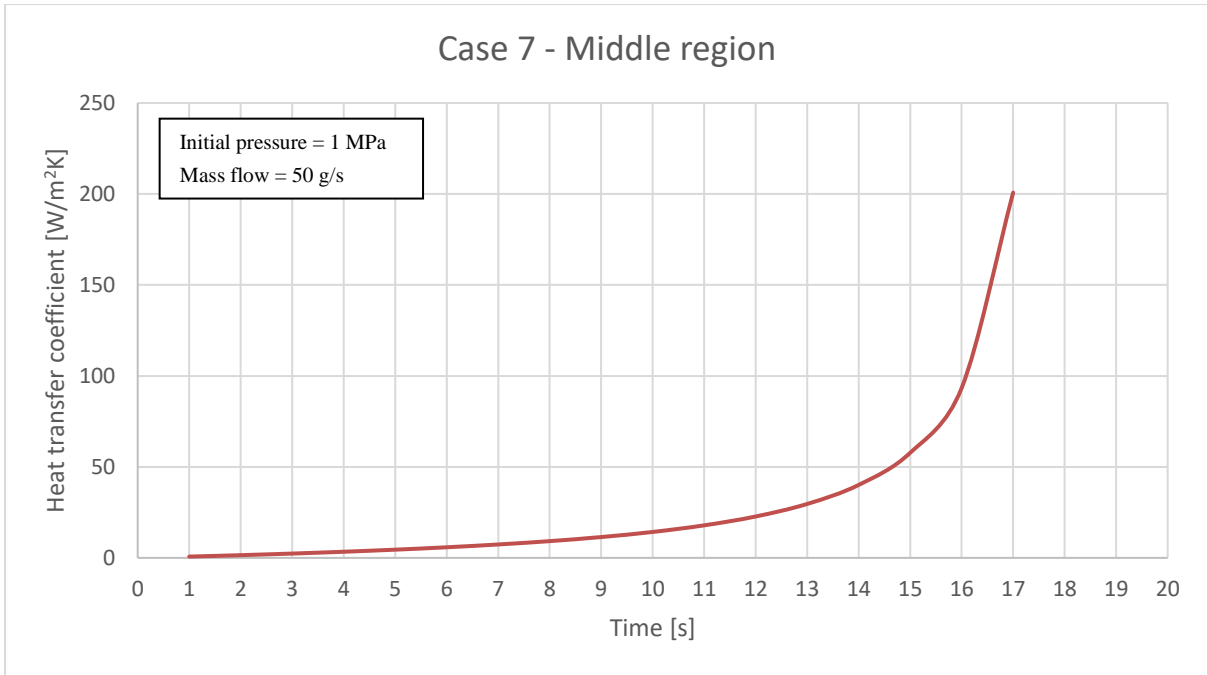
Figure 5.19 Heat transfer coefficient for case 7 at middle region
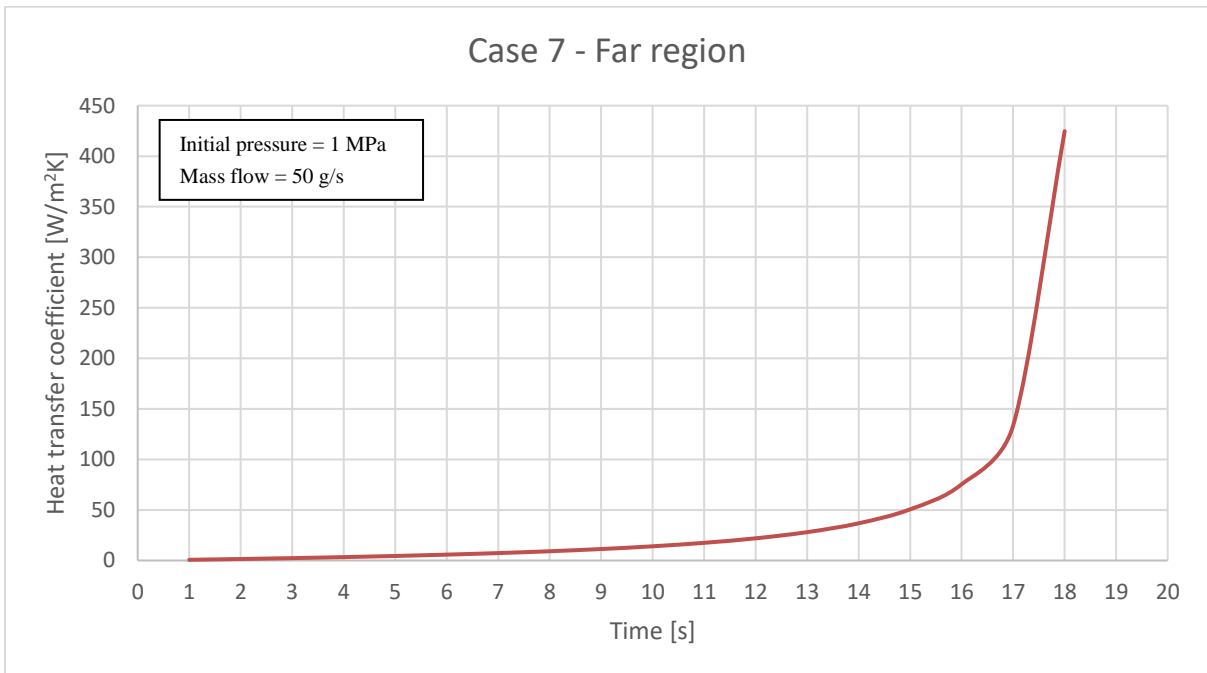


Figure 5.20 Heat transfer coefficient for case 7 at far region

Figure 5.18, Figure 5.19 and Figure 5.20 show the heat transfer coefficient for the tank walls at the inlet, middle and far region for case 7, respectively. The inlet region shows a decrease on heat transfer coefficient due to a lower temperature region generated by the colder inlet gas, resulting in a heat flux to inside the tank. The middle and far tank regions were plotted up to the seventh second as further near the end of the simulation, wall temperature reaches greater values than the reference temperature, resulting in a negative coefficient. This would not be the case in reality, as the reference temperature should not be constant. The three tank

regions show that a higher convective coefficient is developed at the end of the tank, in the far region away from the inlet, with decreasing values towards the inlet region.

## 5.3.1 Literature comparison
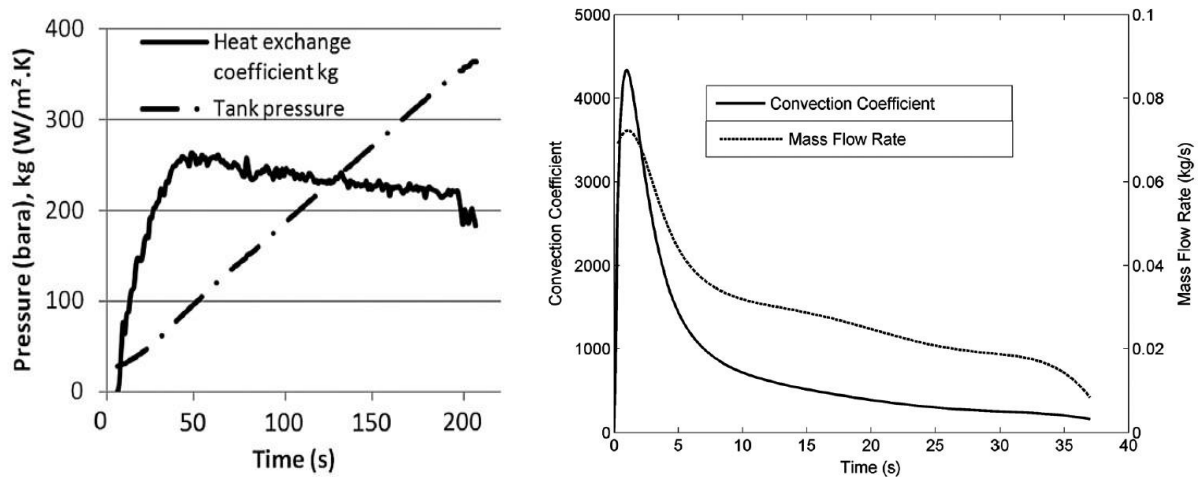


Figure 5.21 Heat transfer coefficient with time [5, 6]

Figure 5.21 are results extracted from the works of Merida et al [5] and Burgeois et al [6], showing similar heat transfer coefficient behavior found in this work. First, the instant change in the convective coefficient occurs rapidly after a few seconds of filling. And second, that the mass flow rate influences the convective coefficient.

As previously discussed, the overall heat transfer coefficient between the wall and gas is heavily influenced by the mass flow rate into the cylinder, achieving higher values with a higher mass flow rate. This is expected as the gas rate flowing into the tank will also have an effect over the magnitude of the velocity and turbulent kinetic energy thus increasing heat transfer at the wall. A precise relationship cannot be determined from the current data obtained, although it is apparent that the mass flow rate and the heat convection coefficient have a substantial correlation.

In simulations with a lower initial pressure condition, a larger heat transfer coefficient is developed on the far and middle region, while smaller values are developed at the inlet region due to the mixing with the colder inlet gas. In some instances, for higher initial pressure condition simulations, negative heat transfer coefficient values are developed, as walls' temperature can go lower than initial ambient temperature condition, leading to a negative heat flux towards the tank. This behavior is unexpected, as no other researched work suggests this can occur. Mesh quality, turbulence model, gravity and buoyancy can have an impact on the heat transfer coefficient. This work neglected gravity and buoyancy effects.

Preliminary simulations showed that a coarser mesh introduces oscillating and unrealistic behavior on the flow field. Since no experimental data is available to validate all the cases for a tank with 0.7 dia. and 6.6 length dimensions, the validity of the results is entirely dependent on the rhoCentralFoam solver's ability to accurately predict the flow.

# 6 Conclusion

The hydrogen gas filling process for a tank with larger dimensions as compared to an on-board vehicle storage tank has been investigated, using openFOAM CFD open-source code. The evolution of the temperature and pressure distribution was evaluated for several cases with different initial pressures. The heat transfer coefficient, even though difficult to determine, was also investigated for the whole tank and at different regions of interest.

The inlet gas temperature was selected as 273.15 K, while the temperature inside the tank, as well as the walls and surroundings were selected as 298.15 K. Two different constant mass flow rates of 10 g/s and 50 g/s at the inlet were assumed. While the initial pressure inside the tank was changed for each of the simulations due to high computational time and time constraints, but with the goal to capture as much of the overall filling process history.

Results show that for both mass flow rates, a lower initial pressure inside the tank will lead a higher rise in temperature. While a higher initial pressure leading to a lower temperature rise. This inversely proportional behavior in the increment in temperature is dependent of the inlet mass flow rate, with a higher mass flow value resulting in more gas being dispensed and compressed inside the tank. Comparing a total rise of temperature of 18.4 degrees for a 50 g/s mass flow and 4.6 degrees for a 10 g/s mass flow, both with a 1 MPa initial pressure.

These findings lead to the possibility of starting the gas refueling process with a low mass flow rate as the instant change in temperature is smaller, and once a higher pressure is developed inside the tank, then increase the mass flow to achieve a faster fill. For instance, a tank with the stated dimensions would hold around 65 Kg of mass at 35 MPa. With a constant mass flow of 10 g/s, the filling would take around 1.8 hours, which may not be reasonable for a filling station. Increasing the mass flow once a higher pressure is reached could lead a significant reduction in filling time.

The simulations suggest that the convective heat transfer coefficient increases steadily through the first seconds of filling. This is in agreement with other research papers. Further development of the convective coefficient was not possible to obtained due to the simulation time restriction. The greatest heat transfer and higher temperature therefore occurs at the end of the tank.

As gravity and buoyancy effects were neglected for this work, their impact on the temperature field and convective transfer coefficient is recommended for further assessment.

## 6.1 Further work

Additional suggestion for further work can be:

- Run simulations of interest in 3-D
- Use a finer mesh or perform mesh dependence analysis
- Update mesh geometry to include tank ends.
- Use and compare different solvers and turbulence models
- Perform experiments to compare and validate the model

In comparison to onboard storage tanks for land vehicles, there appears to be less research on hydrogen storage tanks with larger dimensions, as discussed in the introduction. Therefore, it would be interesting to perform experimental testing on such storage tanks.

# References

[1]     N. de Miguel, B. Acosta, D. Baraldi, R. Melideo, R. Ortiz Cebolla, and P. Moretto, "The role of initial tank temperature on refuelling of on-board hydrogen tanks," *International Journal of Hydrogen Energy,* vol. 41, no. 20, pp. 8606-8615, 2016/06/01/ 2016, doi: https://doi.org/10.1016/j.ijhydene.2016.03.158.

[2]     M. C. Galassi, D. Baraldi, B. Acosta Iborra, and P. Moretto, "CFD analysis of fast filling scenarios for 70 MPa hydrogen type IV tanks," *International Journal of Hydrogen Energy,* vol. 37, no. 8, pp. 6886-6892, 2012/04/01/ 2012, doi: https://doi.org/10.1016/j.ijhydene.2012.01.041.

[3]     M. Hosseini, I. Dincer, G. F. Naterer, and M. A. Rosen, "Thermodynamic analysis of filling compressed gaseous hydrogen storage tanks," *International Journal of Hydrogen Energy,* vol. 37, no. 6, pp. 5063-5071, 2012/03/01/ 2012, doi: https://doi.org/10.1016/j.ijhydene.2011.12.047.

[4]     M. Heitsch, D. Baraldi, and P. Moretto, "Simulation of the Fast Filling of Hydrogen Tanks," 2009. [Online]. Available: https://publications.jrc.ec.europa.eu/repository/handle/JRC51713.

[5]     C. Dicken and W. Mérida, "Modeling the Transient Temperature Distribution within a Hydrogen Cylinder during Refueling," *Numerical Heat Transfer Part A-applications - NUMER HEAT TRANSFER PT A-APPL,* vol. 53, 01/01 2008, doi: 10.1080/10407780701634383.

[6]     T. Bourgeois, F. Ammouri, M. Weber, and C. Knapik, "Evaluating the temperature inside a tank during a filling with highly-pressurized gas," *International Journal of Hydrogen Energy,* vol. 40, no. 35, pp. 11748-11755, 2015/09/21/ 2015, doi: https://doi.org/10.1016/j.ijhydene.2015.01.096.

[7]     Y. Takagi, N. Sugie, K. Takeda, Y. Okano, T. Eguchi, and K. Hirota, "Numerical Investigation of the Thermal Behavior in a Hydrogen Tank During Fast Filling Process," 2011. [Online]. Available: https://doi.org/10.1115/AJTEC2011-44270.

[8]     M. Liszka, A. Fridlyand, A. Jayaraman, M. Bonnema, and C. Sishtla, *CFD Modeling of the Hydrogen Fast Filling Process for Type 3 Cylinders and Cylinders Lined With Phase Change Material*. 2019.

[9]     H. K. Versteeg and W. Malalasekera, *An Introduction to Computational Fluid Dynamics: The Finite Volume Method*. Pearson Education Limited, 2007.

[10]    D.-Y. Peng and D. B. Robinson, "A New Two-Constant Equation of State," *Industrial & Engineering Chemistry Fundamentals,* vol. 15, no. 1, pp. 59-64, 1976/02/01 1976, doi: 10.1021/i160057a011.

[11]    D. W. Hahn and M. N. Özisik, *Heat Conduction*. Wiley, 2012.

[12]    K. Mooney, T. Maric, and J. Höpken, *The OpenFOAM Technology Primer*. 2021.

[13]    ANSYS, "6.3.2.1.3. Turbulence Intensity," in *ANSYS Fluent User's Guide*. Canonsburg, PA 15317: ANSYS, Inc, 2013.

[14]    A. Kurganov and E. Tadmor, "New High-Resolution Central Schemes for Nonlinear Conservation Laws and Convection–Diffusion Equations," *Journal of Computational*

*Physics,* vol. 160, no. 1, pp. 241-282, 2000/05/01/ 2000, doi: https://doi.org/10.1006/jcph.2000.6459.

[15]     C. Greenshields. "OpenFOAM v8 User Guide." https://cfd.direct/openfoam/user-guide/v8-fvschemes/ (accessed 12th of April, 2021).

[16]     J. R. Andrews and O. Biblarz, "Temperature dependence of gas properties in polynomial form," Naval Postgraduate School, Monterey, CA 93940, 1981. Accessed: 20th of April 2021. [Online]. Available: https://core.ac.uk/download/pdf/36723064.pdf

[17]     R. D. McCarty, J. Hord, and H. M. Roder, "Selected Properties of Hydrogen (Engineering Design Data)," Center for Chemical Engineering, National Engineering Laboratory, National Bureau of Standards, Boulder, CO 80303, 1981. Accessed: 20th of April 2021. [Online]. Available: https://nvlpubs.nist.gov/nistpubs/Legacy/MONO/nbsmonograph168.pdf

[18]     W. G. Don and H. P. Robert, *Perry's Chemical Engineers' Handbook, Eighth Edition*, 8th ed. / ed. New York: McGraw-Hill Education (in en), 2008.

[19]     HydrogenTools. "Hydrogen Viscosity at different temperatures and pressures | Hydrogen Tools." https://h2tools.org/hyarc/hydrogen-data/hydrogen-viscosity-different-temperatures-and-pressures (accessed 17th of May, 2021).

[20]     HydrogenTools. "Hydrogen Density at different temperatures and pressures | Hydrogen Tools." https://h2tools.org/hyarc/hydrogen-data/hydrogen-density-different-temperatures-and-pressures (accessed 17th of May, 2021).

[21]     HydrogenTools. "Hydrogen Compressibility at different temperatures and pressures | Hydrogen Tools." https://h2tools.org/hyarc/hydrogen-data/hydrogen-compressibility-different-temperatures-and-pressures (accessed 17th of May, 2021).

# Appendices

## Appendix A. Results

Result plots are shown in this appendix.

## A.1. Tank temperature, pressure, and overall heat transfer coefficient.

### A.1.1. Case 1

Table A-1 Case 1 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|:---:|:---:|:---:|
| 273.15 | 10 | 1 |



Figure A-1 Case 1 average temperature and pressure

Figure A-2 Case 1 overall heat transfer coefficient for all regions



Figure A-3 Case 1 overall heat transfer coefficient for inlet region

Figure A-4 Case 1 overall heat transfer coefficient for middle region



Figure A-5 Case 1 overall heat transfer coefficient for far region

## A.1.2. Case 2

Table A-2 Case 2 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 10 | 5 |

Figure A-6 Case 2 average temperature and pressure



Figure A-7 Case 2 overall heat transfer coefficient for all regions

Figure A-8 Case 2 overall heat transfer coefficient for inlet region



Figure A-9 Case 2 overall heat transfer coefficient for middle region

Figure A-10 Case 2 overall heat transfer coefficient for far region

### A.1.3.    Case 3

Table A-3 Case 3 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 10 | 10 |



Figure A-11 Case 3 average temperature and pressure

Figure A-12 Case 3 overall heat transfer coefficient for all regions



Figure A-13 Case 3 overall heat transfer coefficient for inlet region

Figure A-14 Case 3 overall heat transfer coefficient for middle region



Figure A-15 Case 3 overall heat transfer coefficient for far region

## A.1.4.　　　Case 4

Table A-4 Case 4 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 10 | 20 |

Figure A-16 Case 4 average temperature and pressure



Figure A-17 Case 4 overall heat transfer coefficient for all regions

Figure A-18 Case 4 overall heat transfer coefficient for inlet region



Figure A-19 Case 4 overall heat transfer coefficient for middle region

Figure A-20 Case 4 overall heat transfer coefficient for far region

## A.1.5.       Case 5

Table A-5 Case 5 initial and boundary conditions

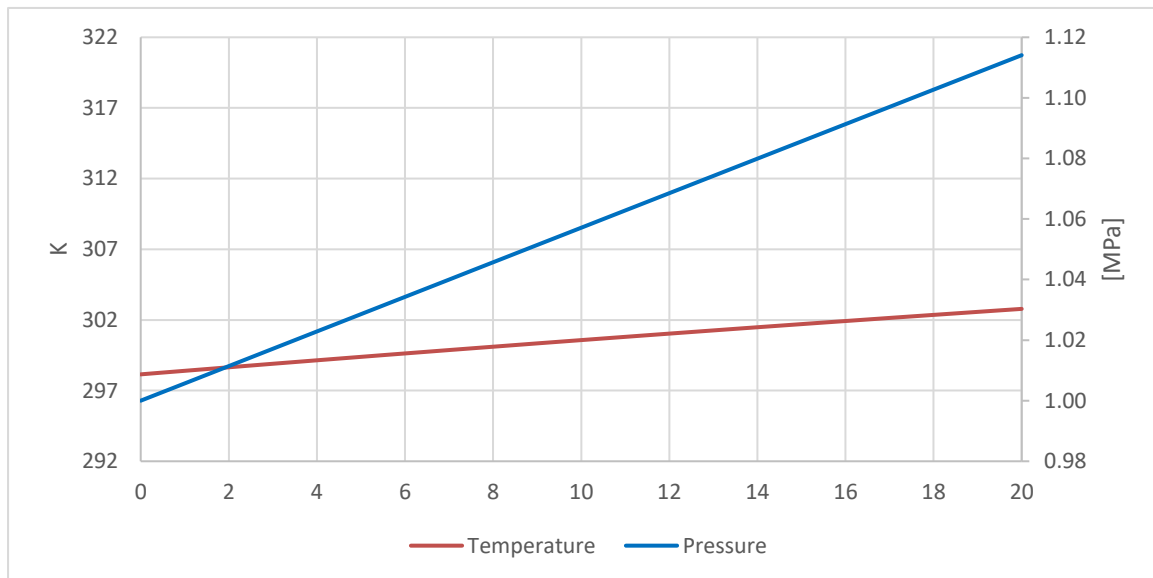| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 10 | 30 |



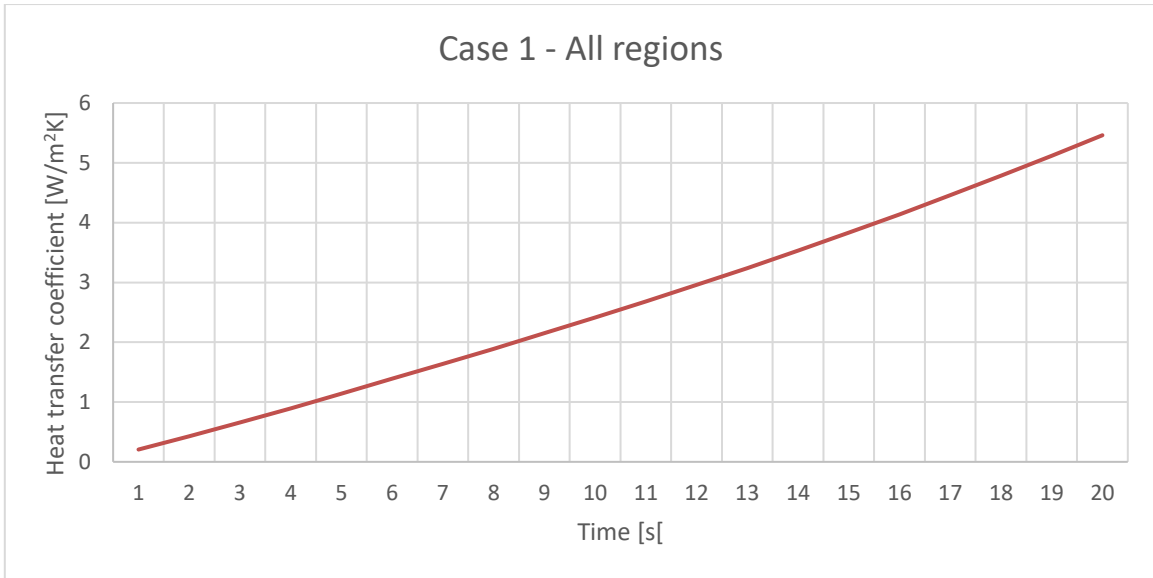Figure A-21 Case 5 average temperature and pressure

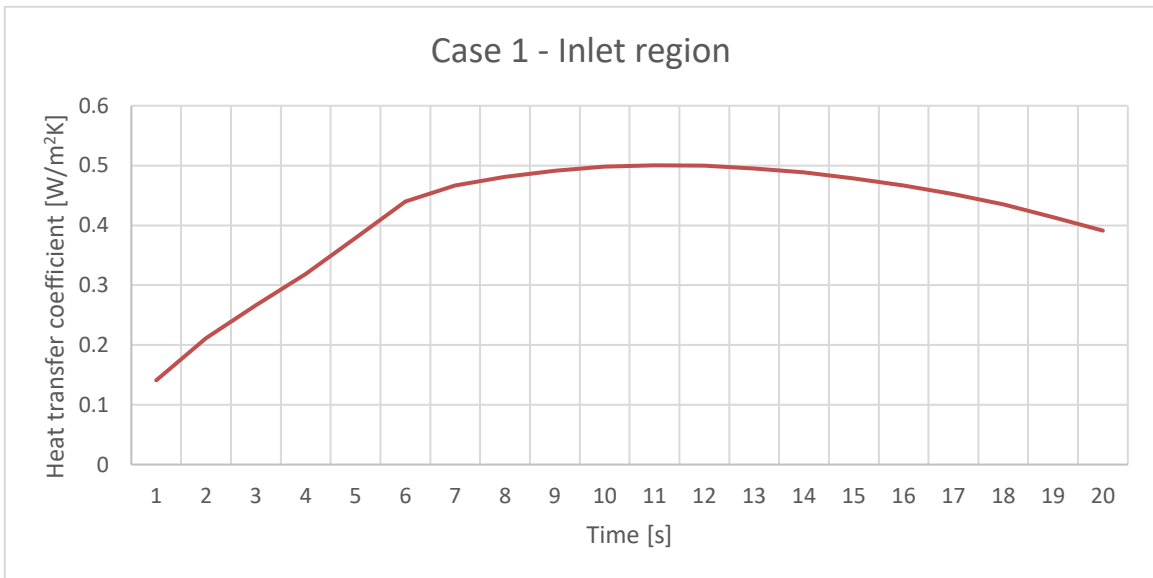Figure A-22 Case 5 overall heat transfer coefficient for all regions



Figure A-23 Case 5 overall heat transfer coefficient for inlet region

Figure A-24 Case 5 overall heat transfer coefficient for middle region



Figure A-25 Case 5 overall heat transfer coefficient for far region

## A.1.6.      Case 6

Table A-6 Case 6 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 10 | 35 |

Figure A-26 Case 6 average temperature and pressure
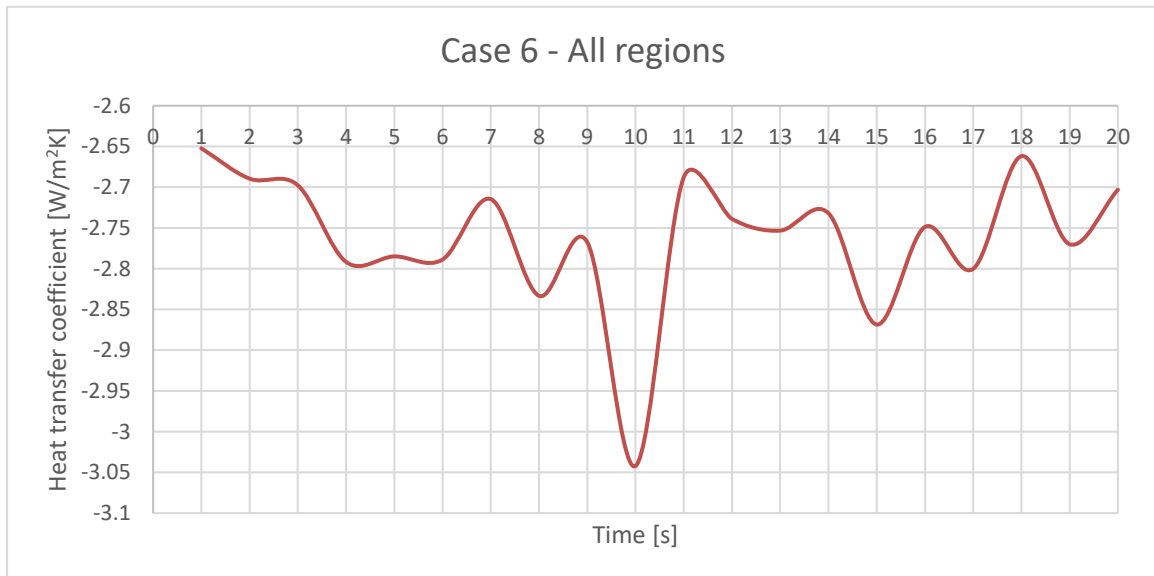


Figure A-27 Case 6 overall heat transfer coefficient for all regions
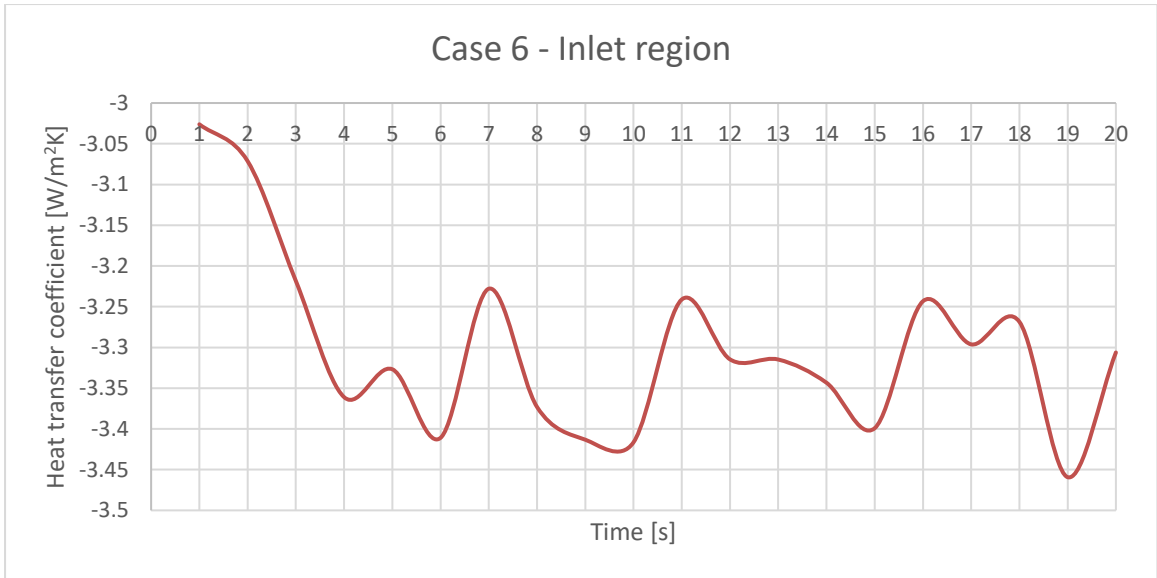
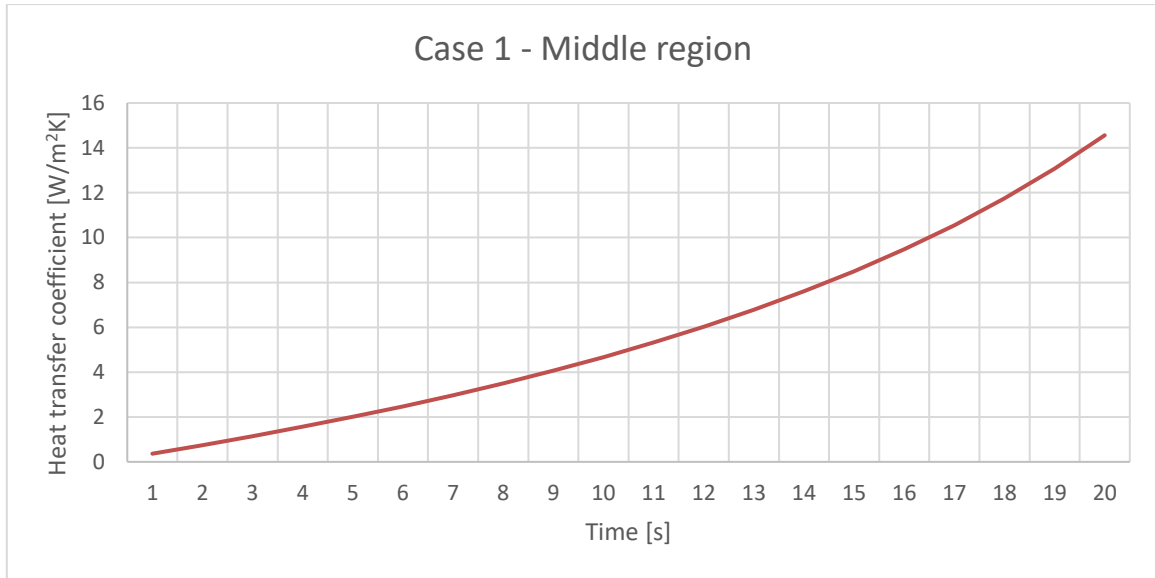Figure A-28 Case 6 overall heat transfer coefficient for inlet region
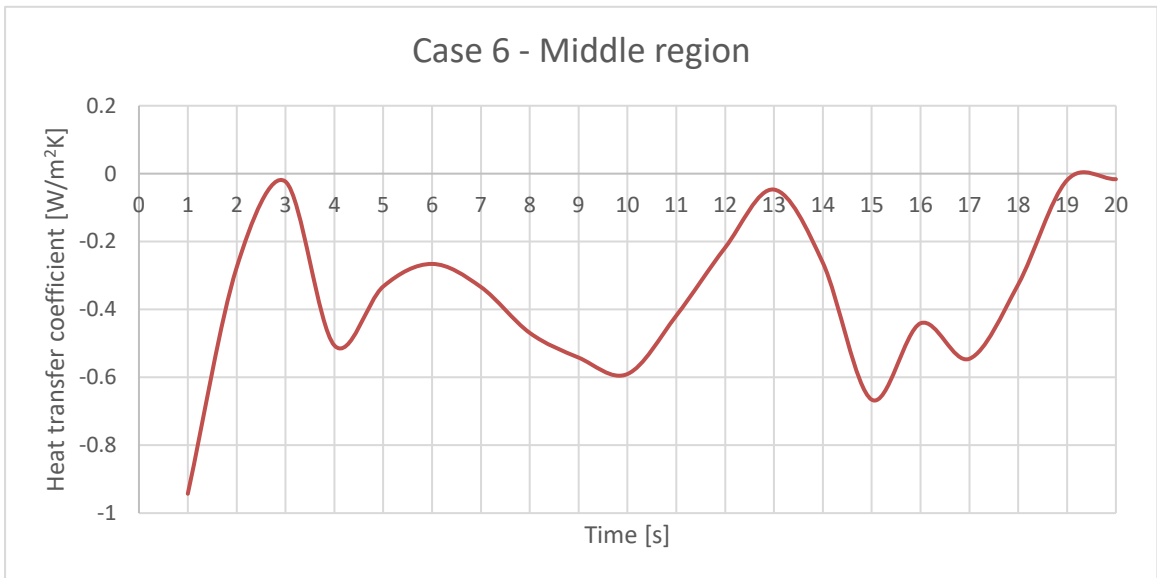


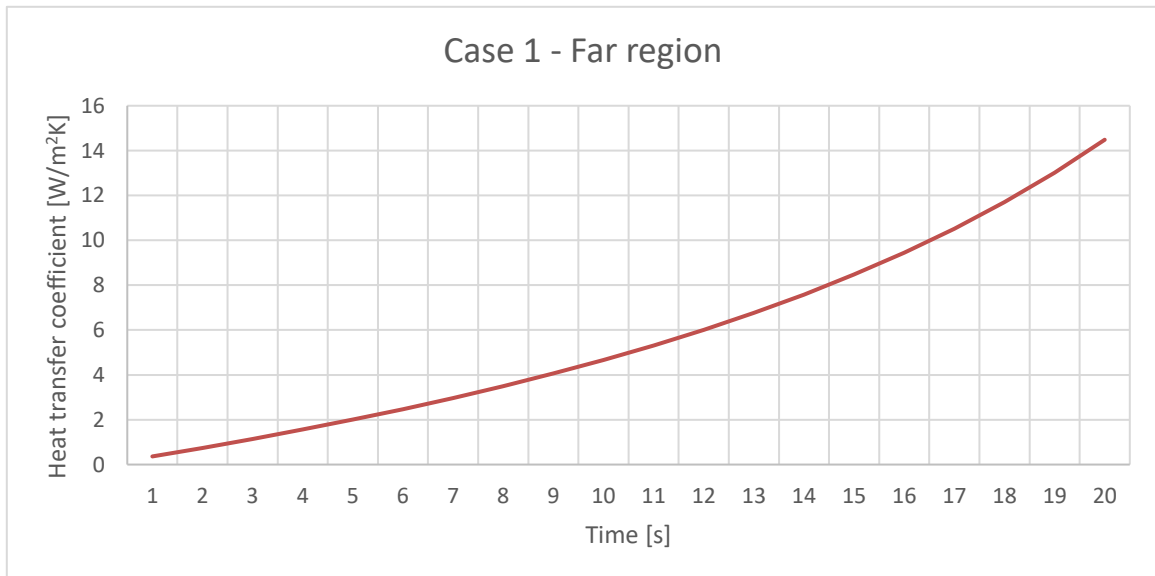Figure A-29 Case 6 overall heat transfer coefficient for middle region
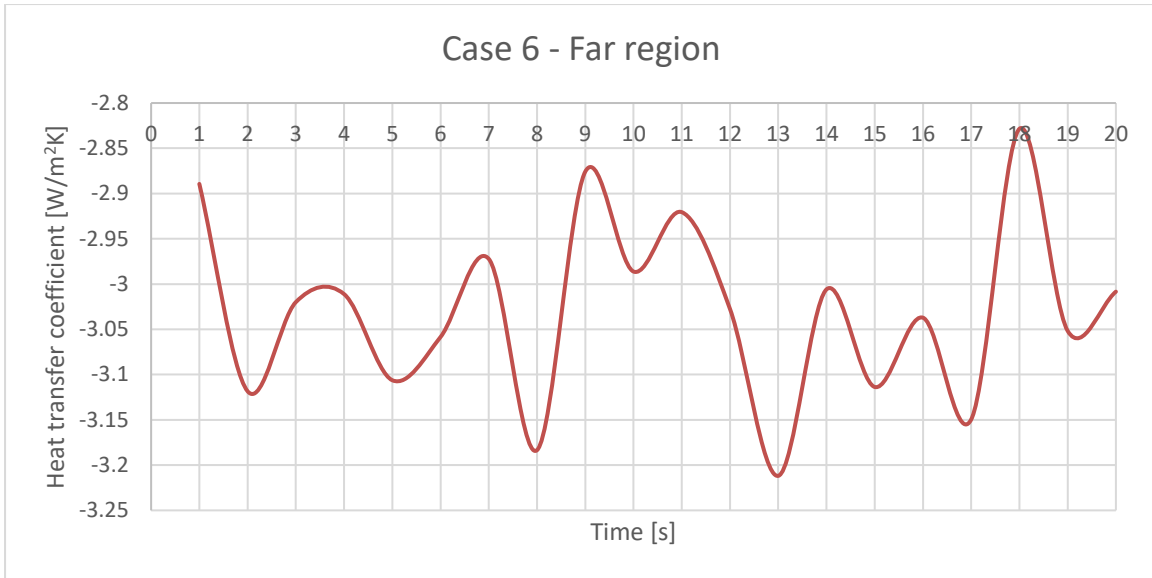
Figure A-30 Case 6 overall heat transfer coefficient for far region

### A.1.7.   Case 7

Table A-7 Case 7 initial and boundary conditions

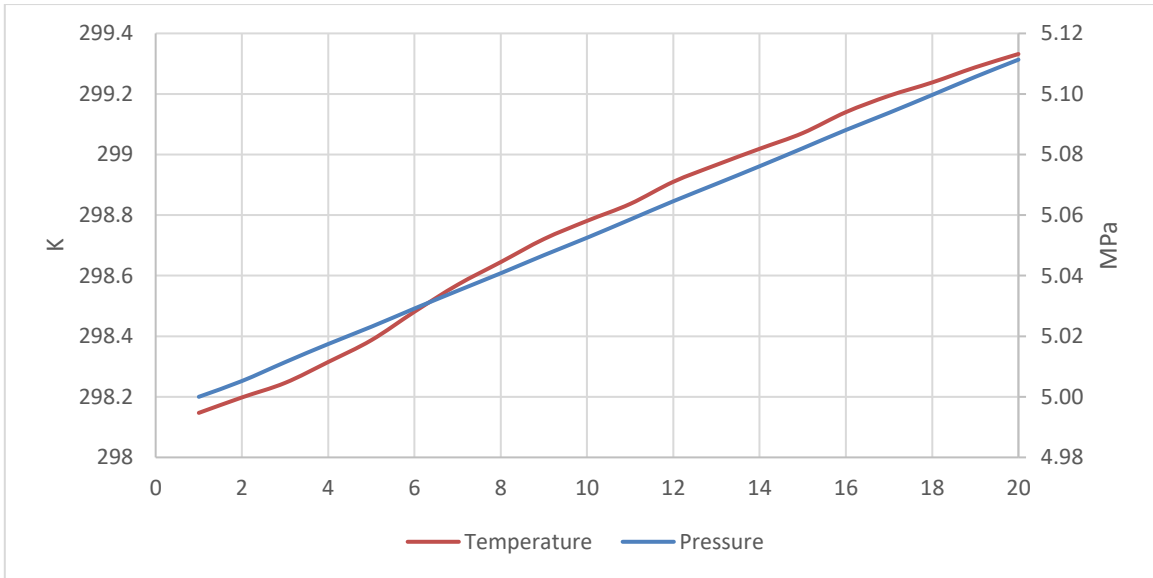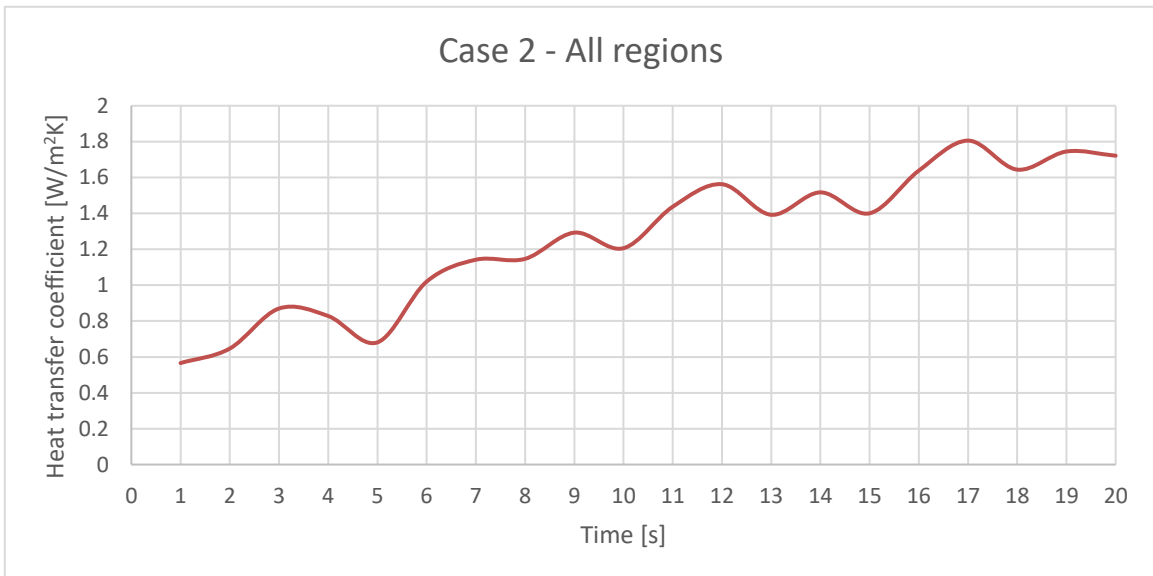| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 1 |
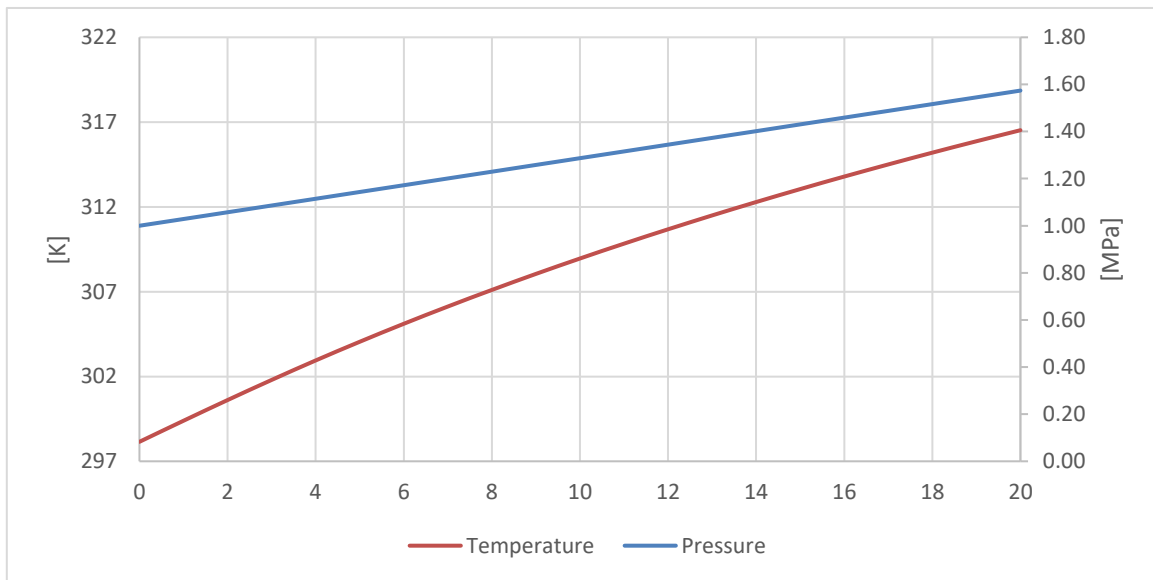


Figure A-31 Case 7 average temperature and pressure
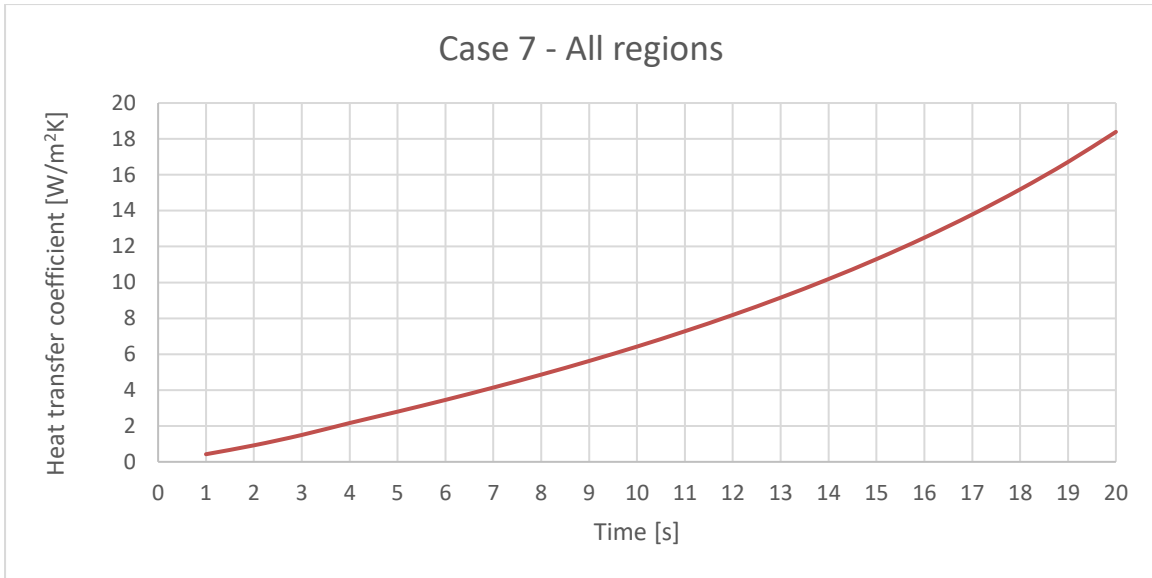
Figure A-32 Case 7 overall heat transfer coefficient for all regions
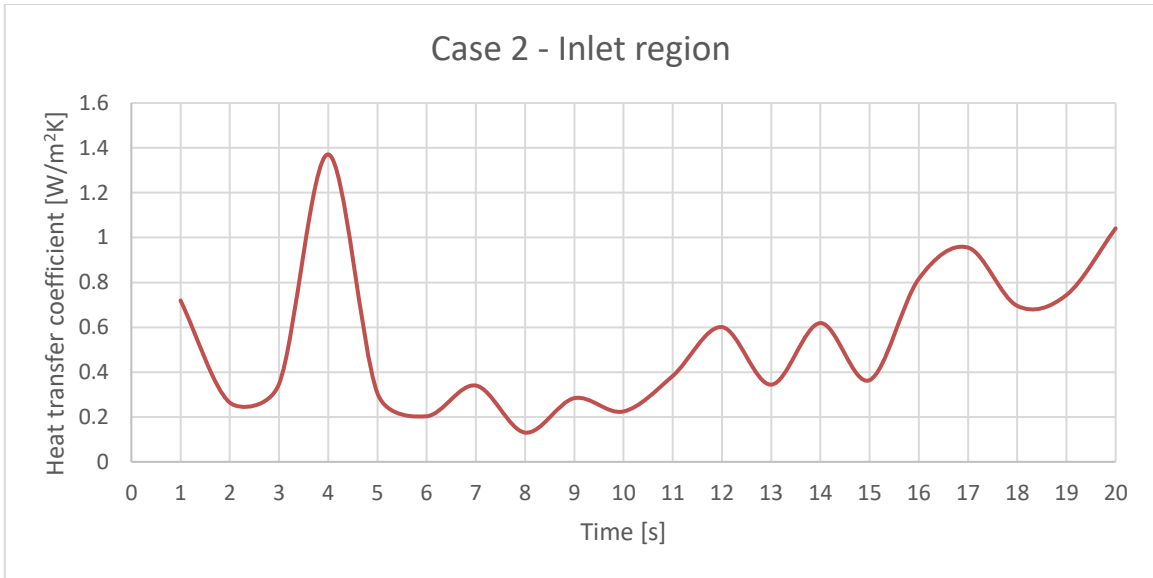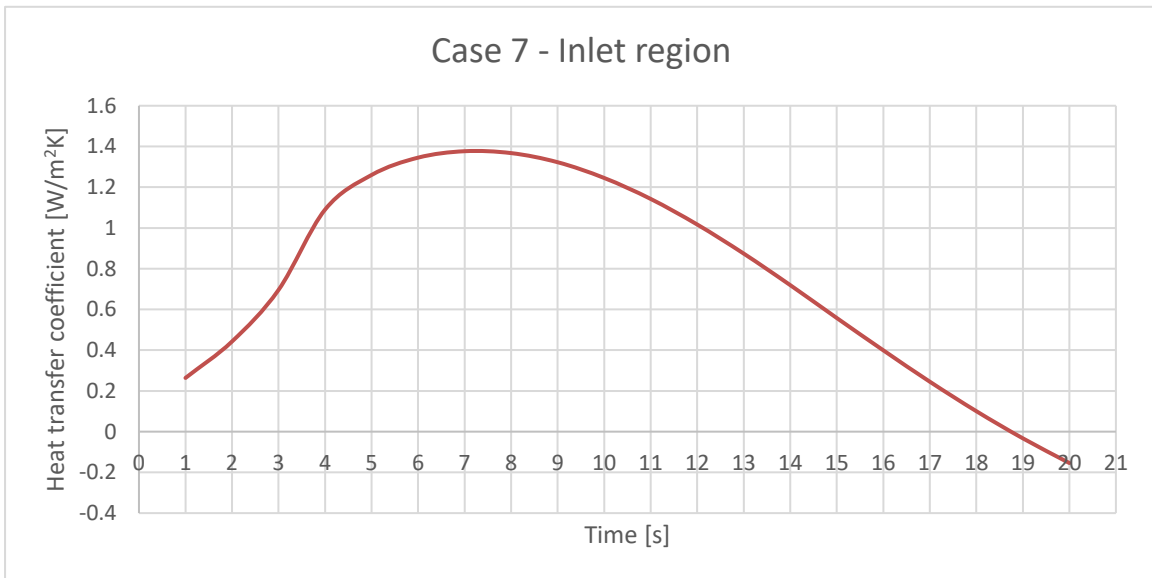


Figure A-33 Case 7 overall heat transfer coefficient for inlet region
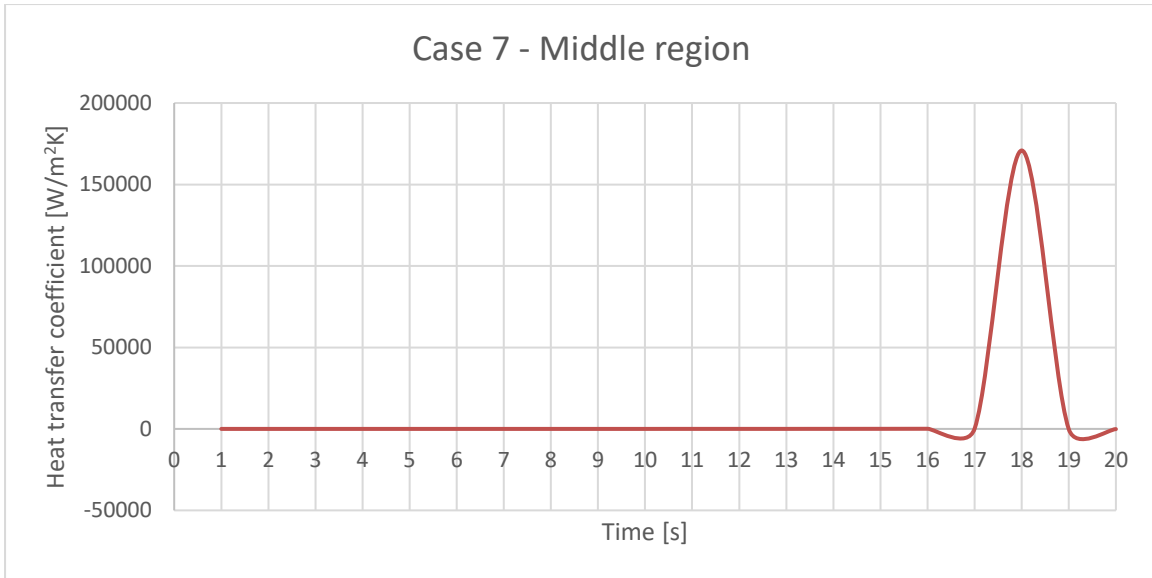
Figure A-34 Case 7 overall heat transfer coefficient for middle region
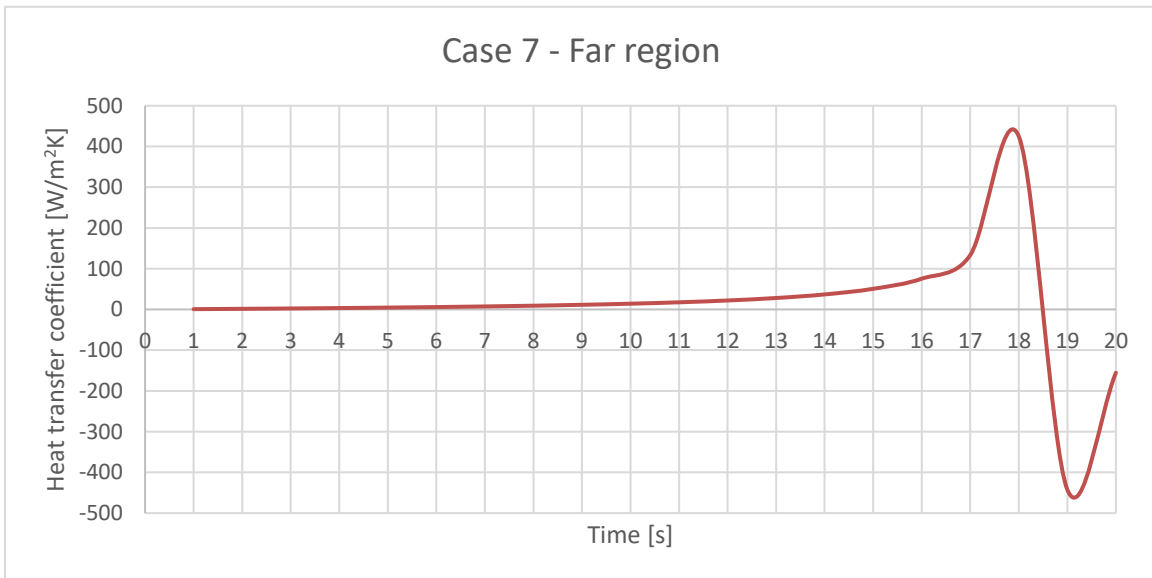


Figure A-35 Case 7 overall heat transfer coefficient for far region

## A.1.8. Case 8

Table A-8 Case 8 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 5 |
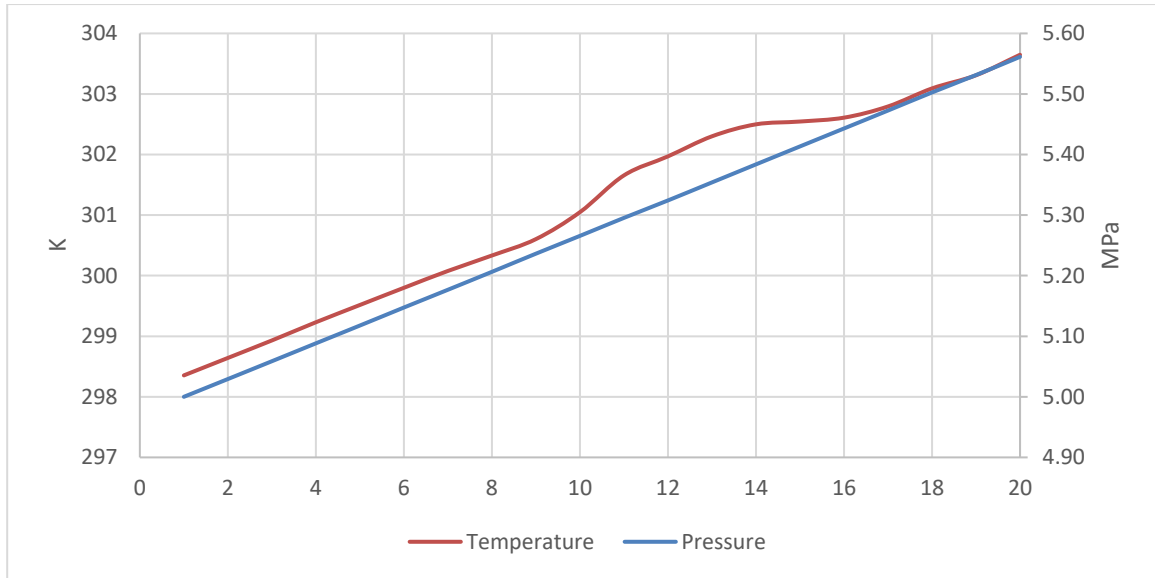
Figure A-36 Case 8 average temperature and pressure
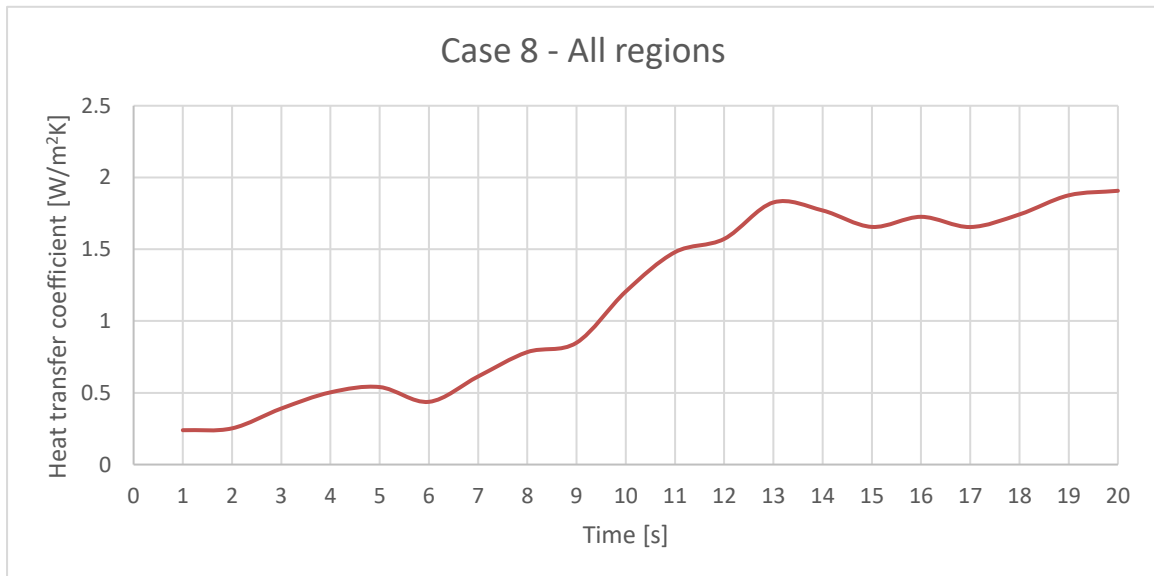


Figure A-37 Case 8 overall heat transfer coefficient for all regions
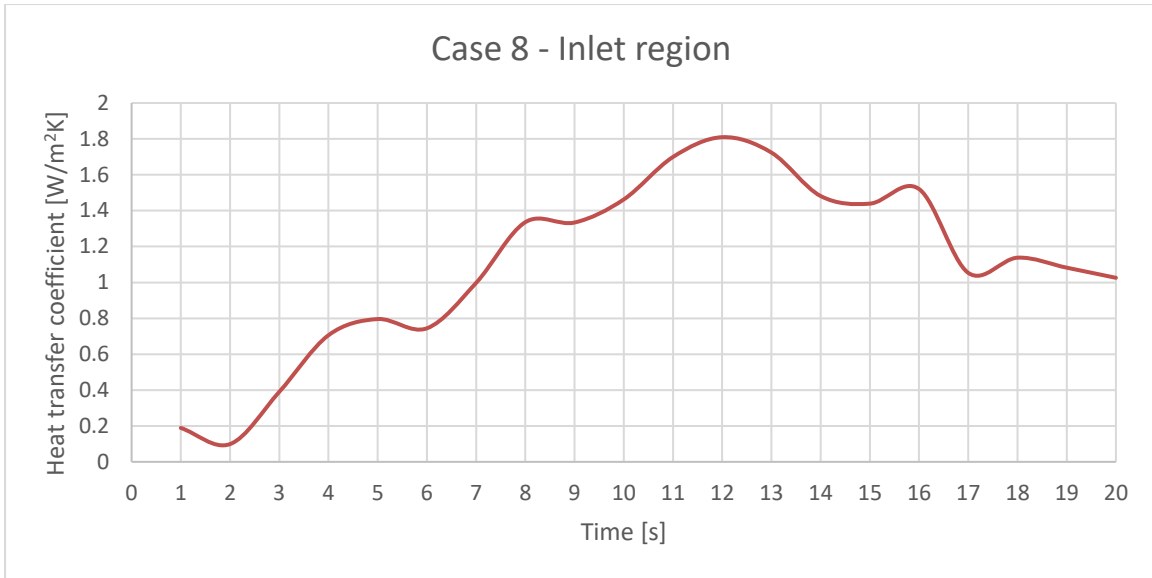
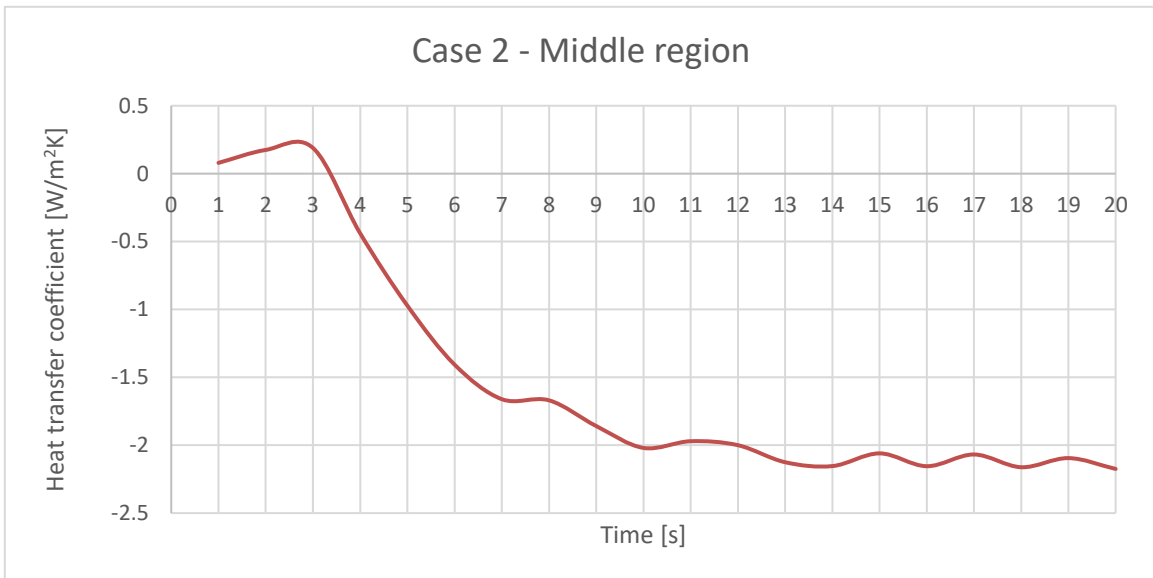Figure A-38 Case 8 overall heat transfer coefficient for inlet region
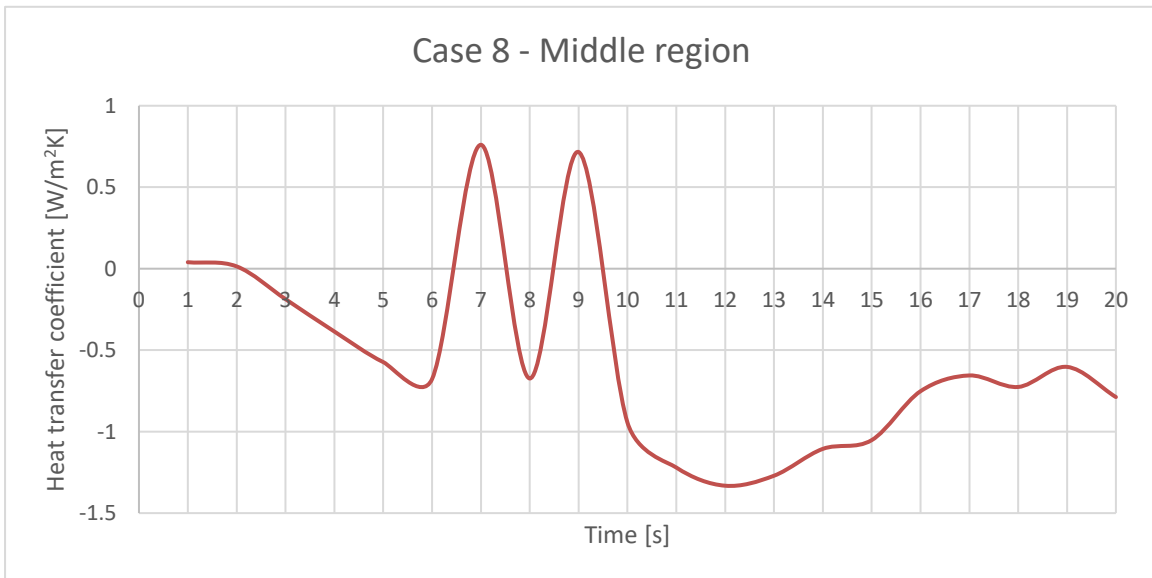


Figure A-39 Case 8 overall heat transfer coefficient for middle region
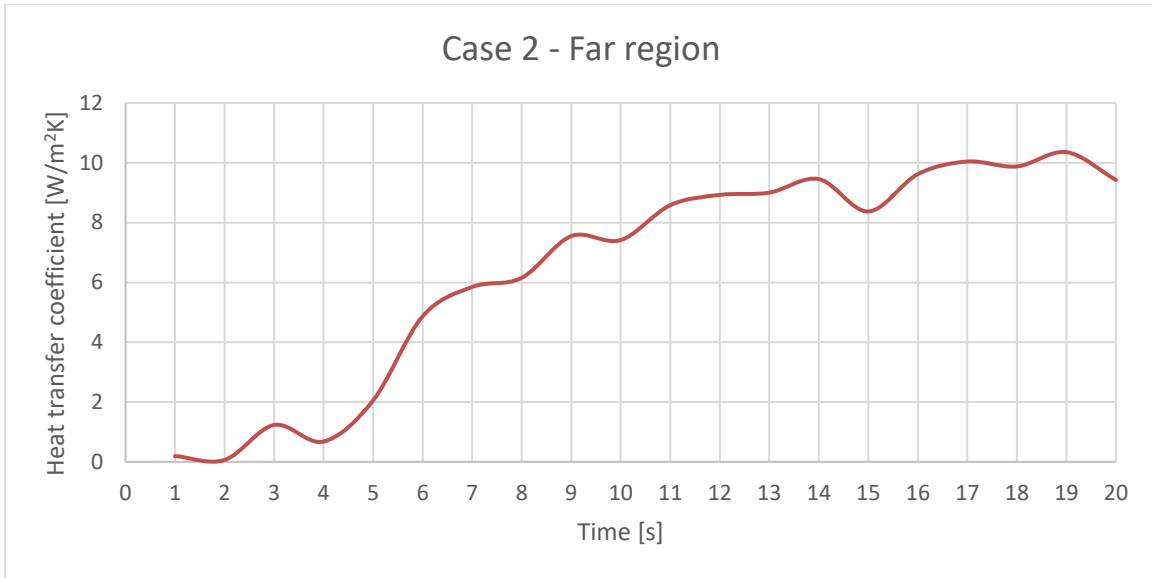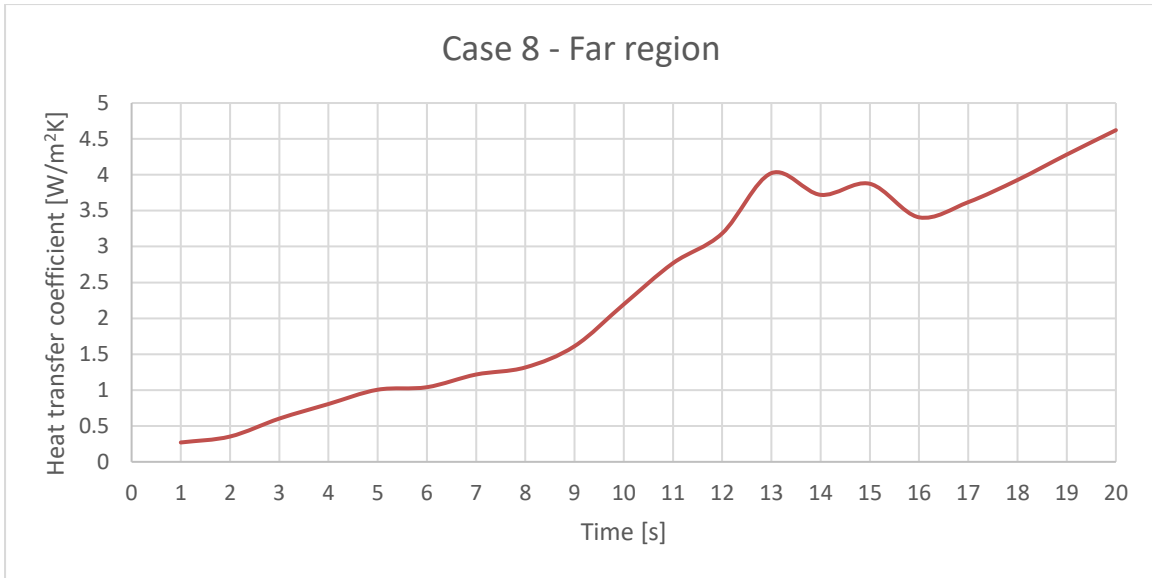
Figure A-40 Case 8 overall heat transfer coefficient for far region

## A.1.9.      Case 9

Table A-9 Case 9 initial and boundary conditions

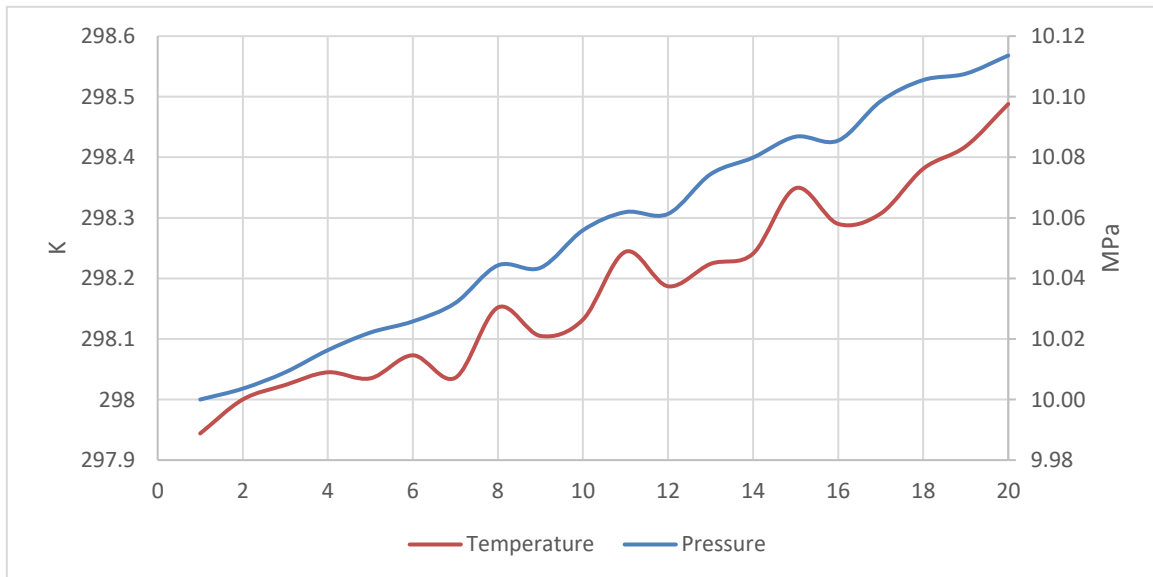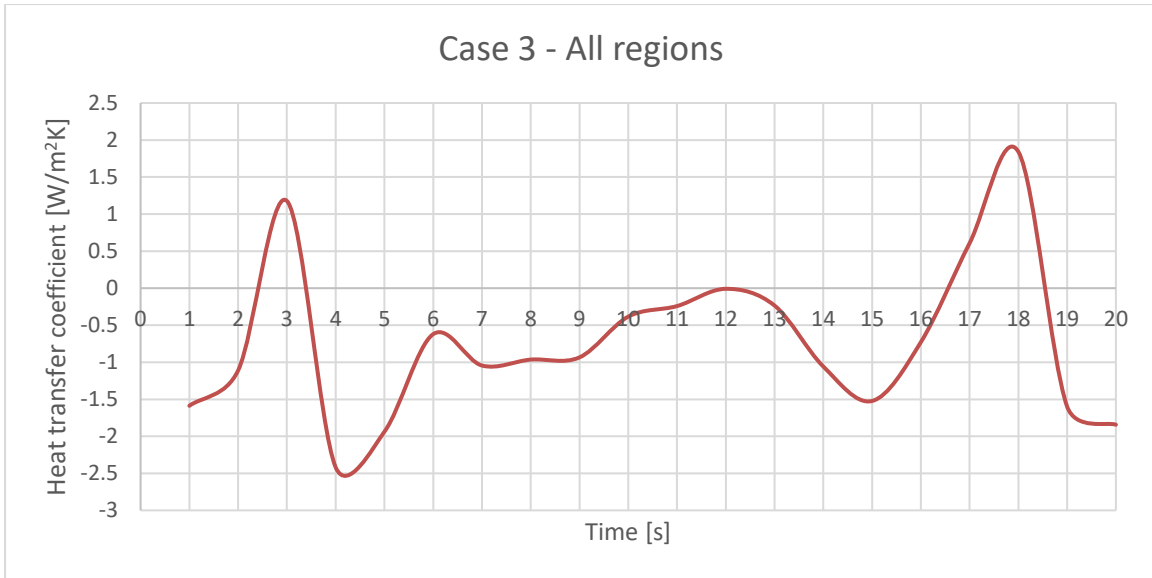| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 10 |
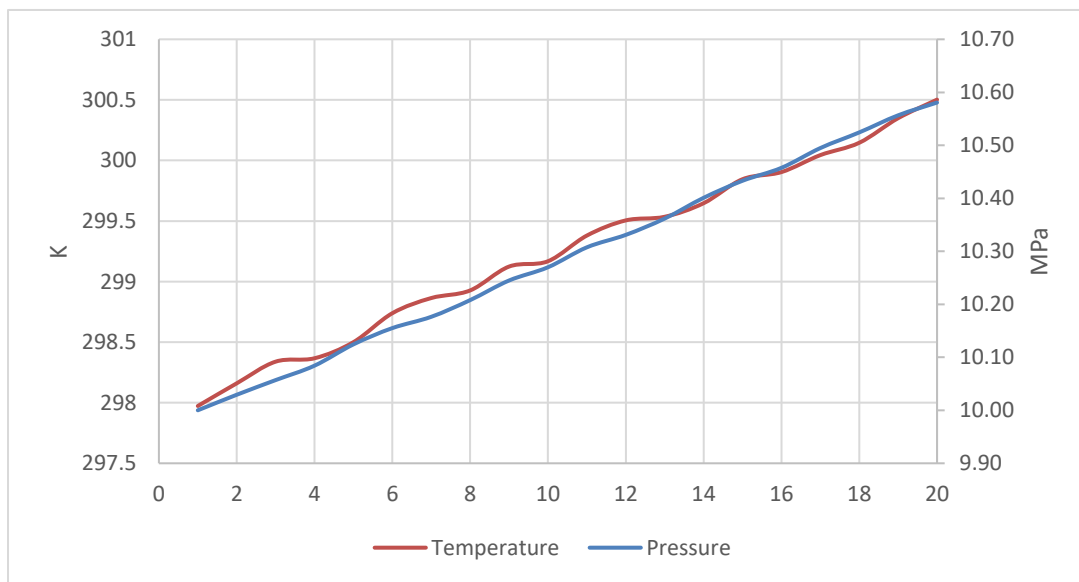


Figure A-41 Case 9 average temperature and pressure
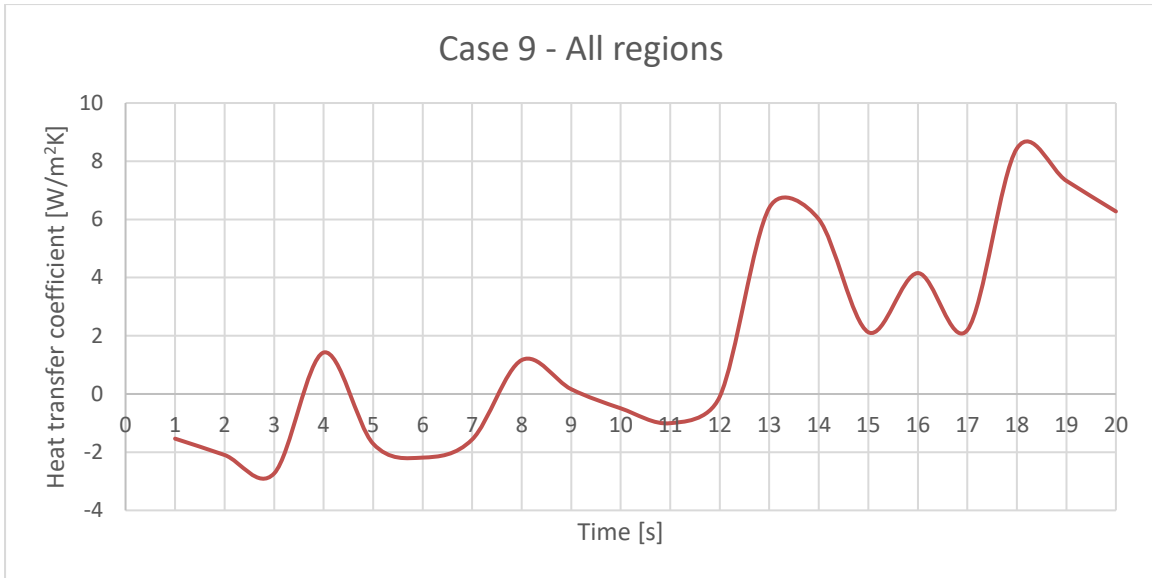
Figure A-42 Case 9 overall heat transfer coefficient for all regions
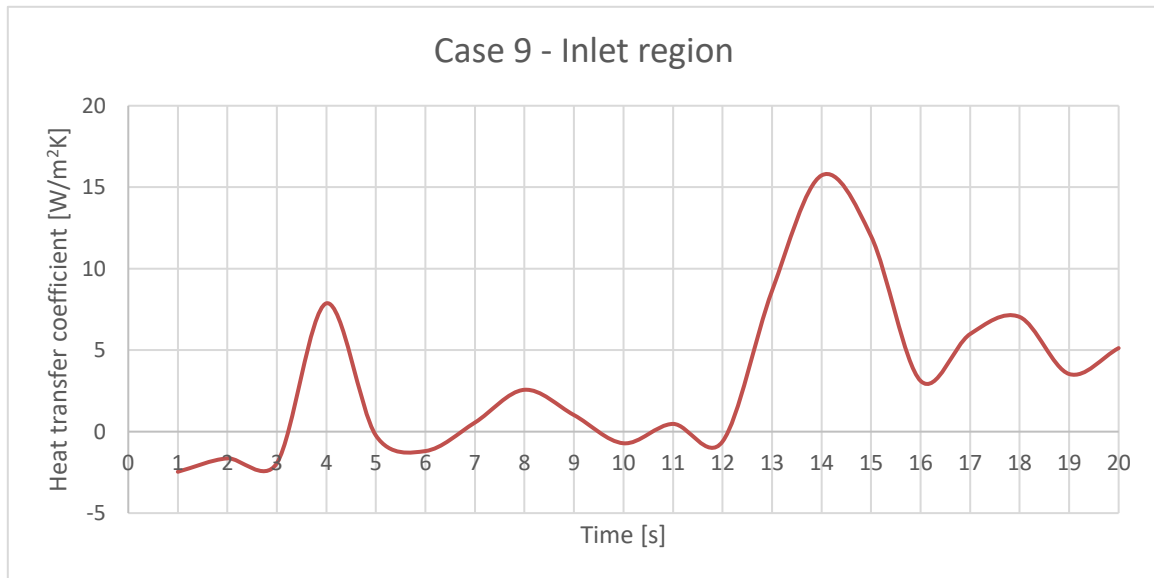


Figure A-43 Case 9 overall heat transfer coefficient for inlet region
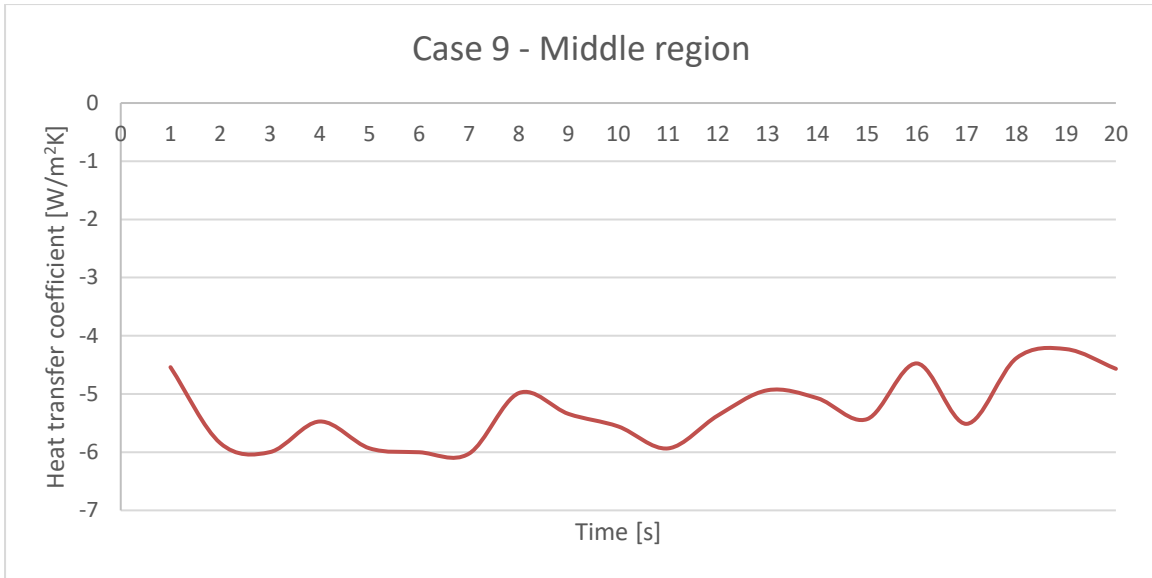
Figure A-44 Case 9 overall heat transfer coefficient for middle region
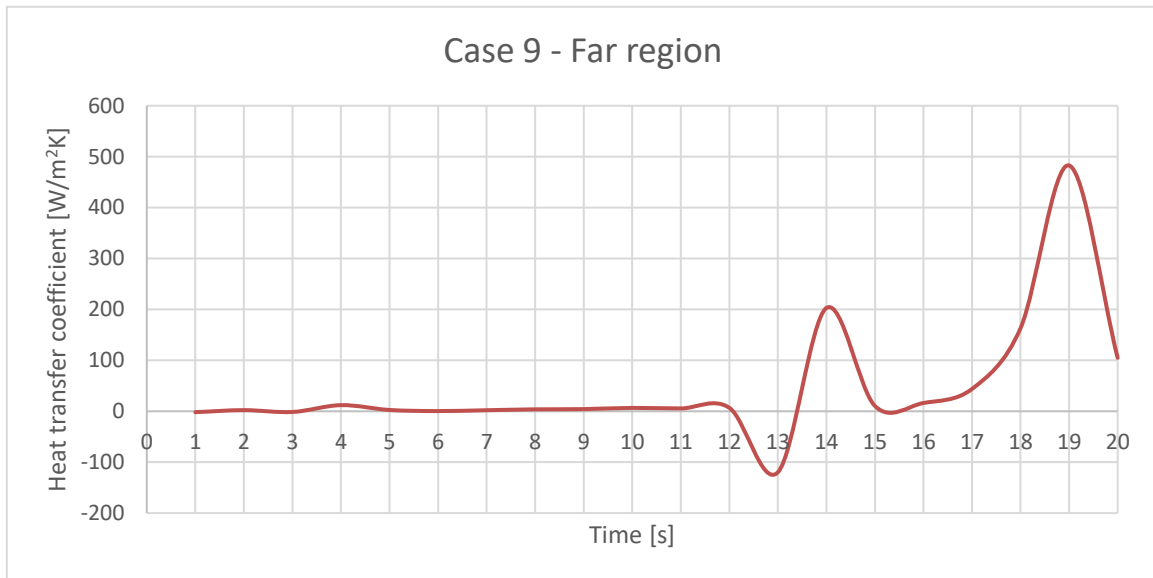


Figure A-45 Case 9 overall heat transfer coefficient for far region

## A.1.10.    Case 10

Table A-10 Case 10 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 20 |
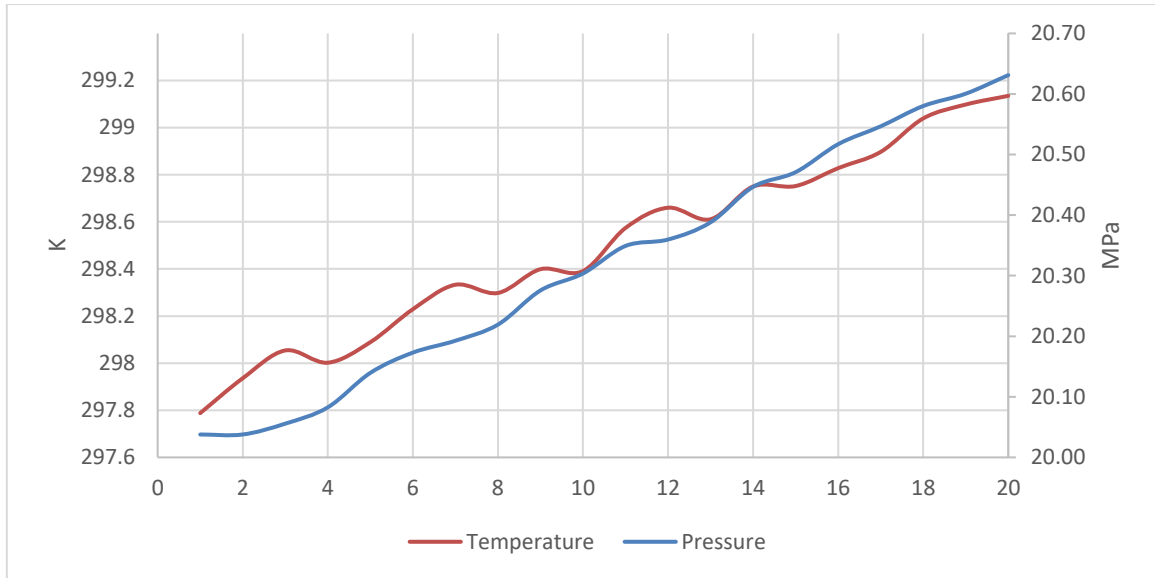
Figure A-46 Case 10 average temperature and pressure
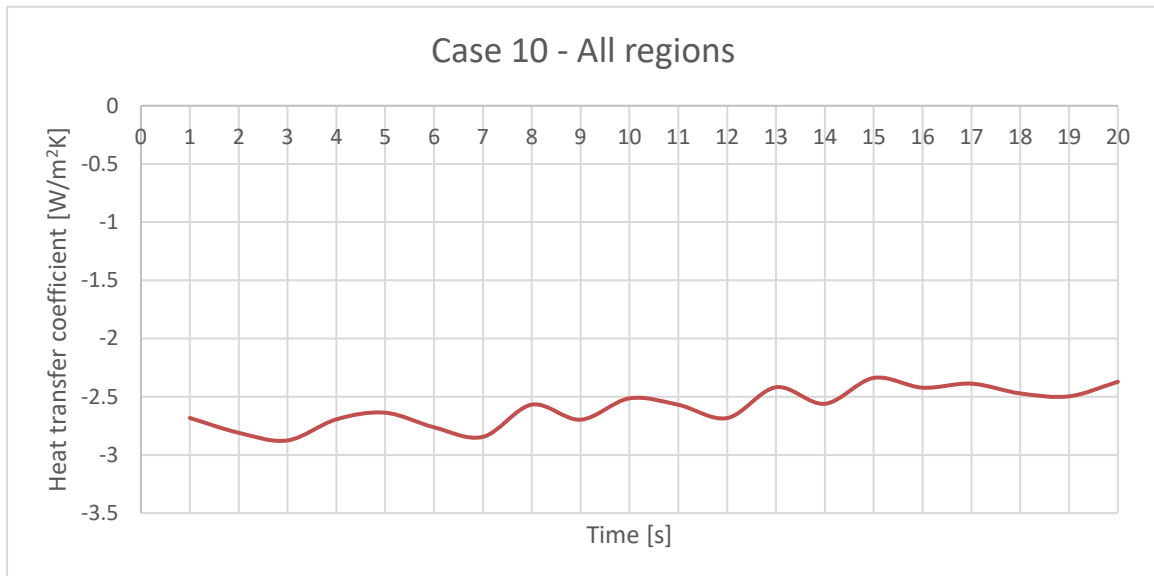


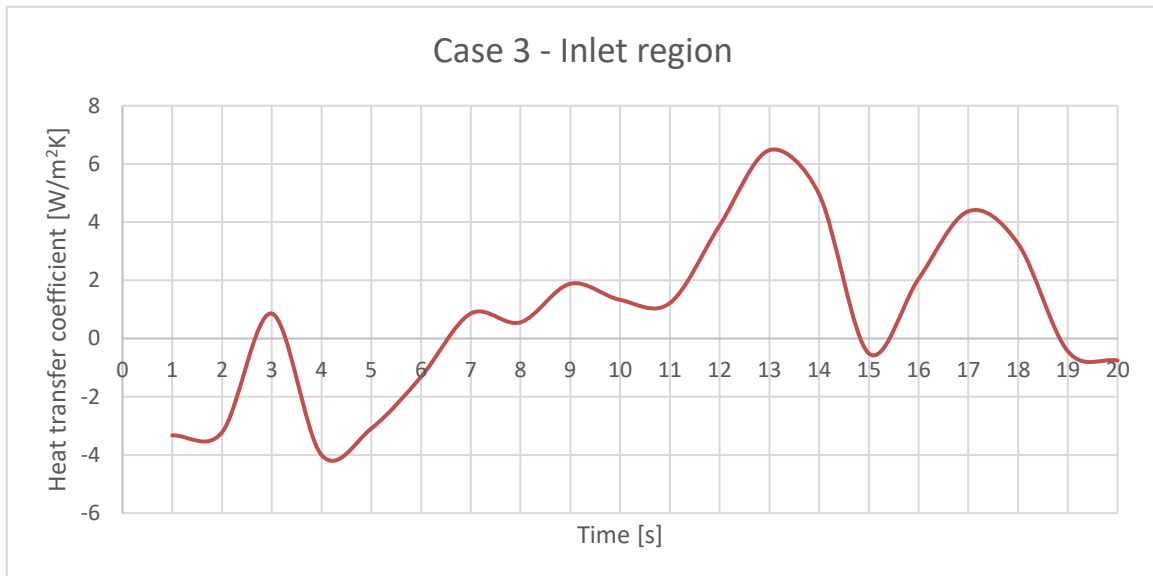Figure A-47 Case 10 overall heat transfer coefficient for all regions
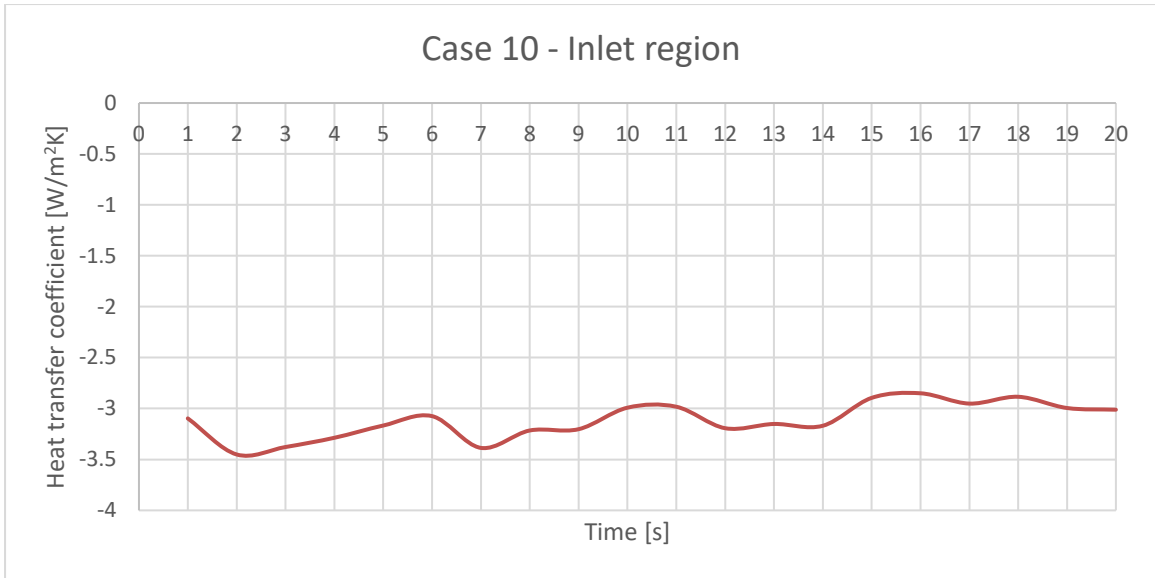
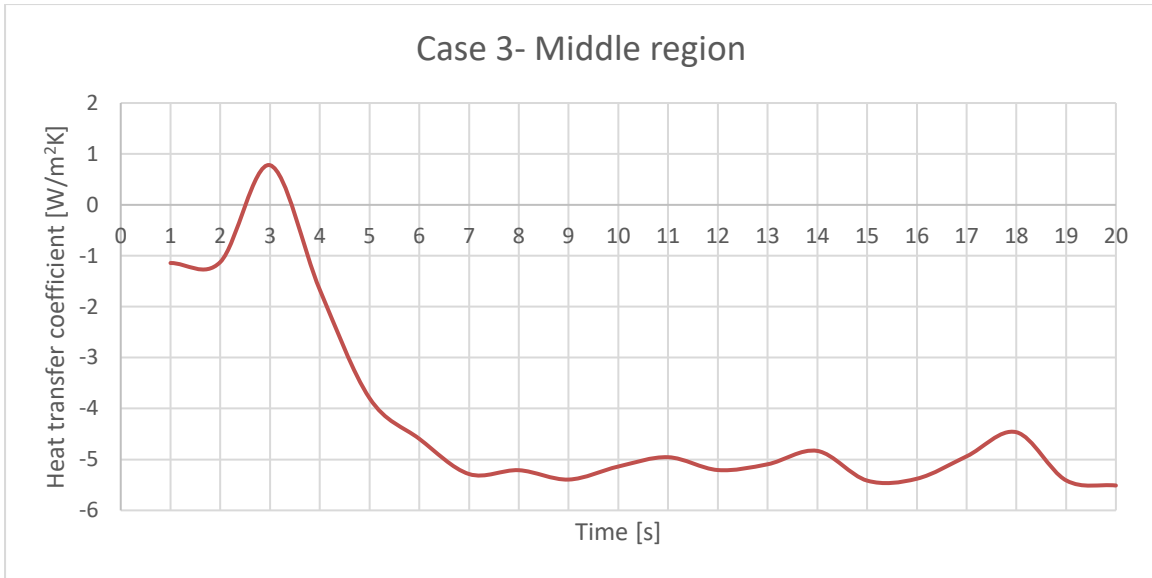Figure A-48 Case 10 overall heat transfer coefficient for inlet region
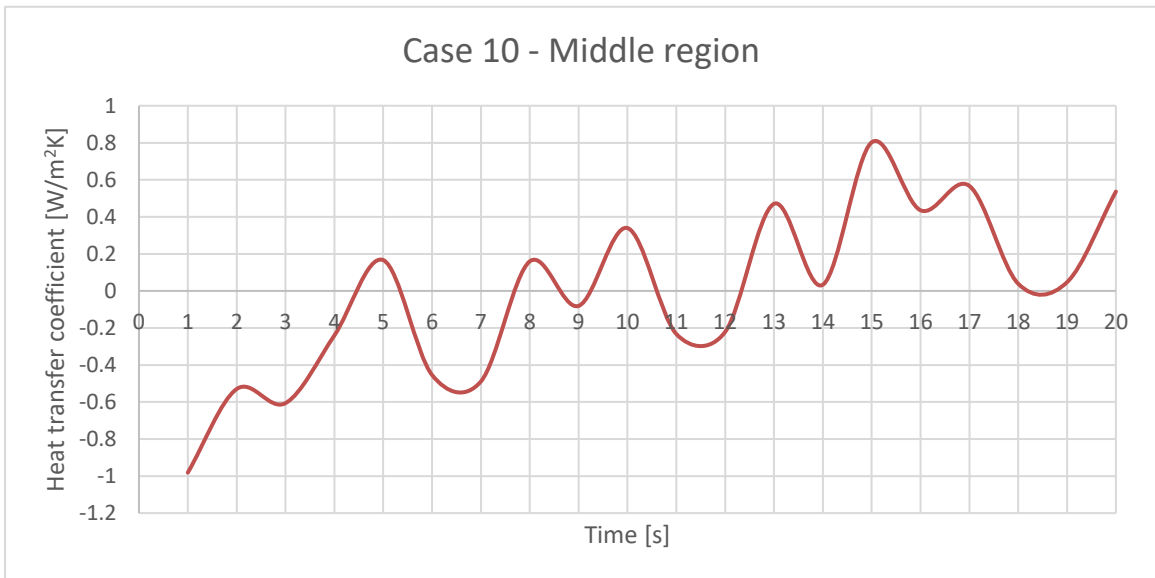


Figure A-49 Case 10 overall heat transfer coefficient for middle region
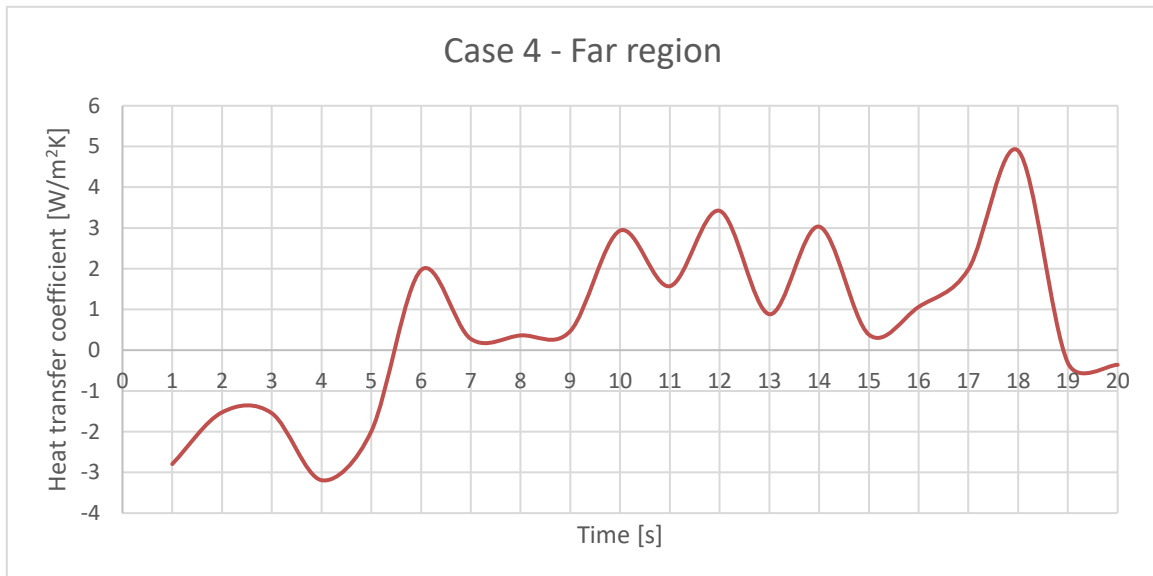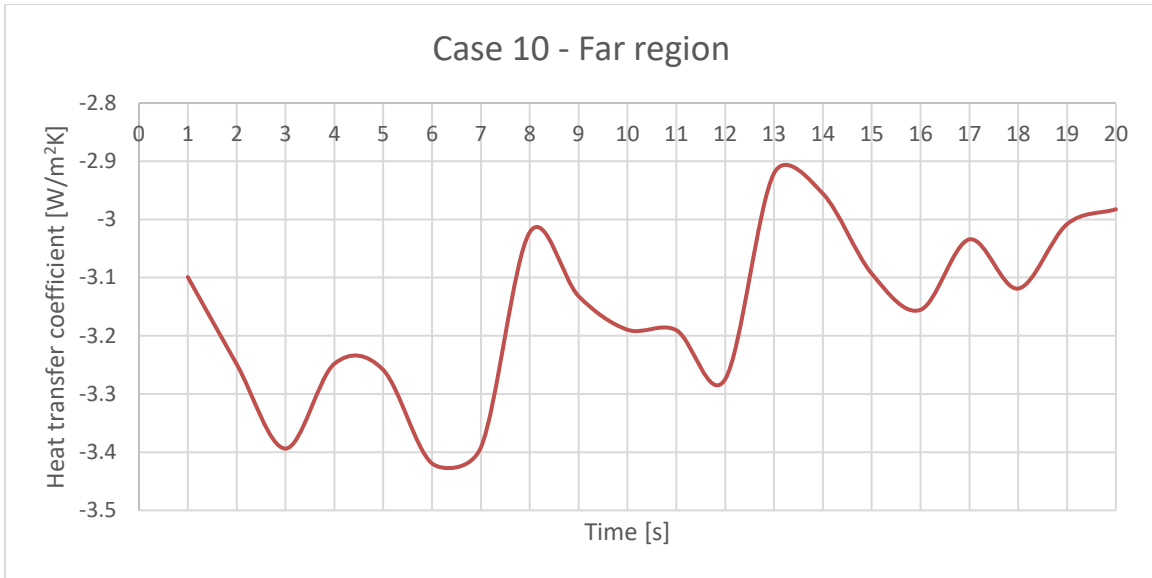
Figure A-50 Case 10 overall heat transfer coefficient for far region

### A.1.11. Case 11

Table A-11 Case 11 initial and boundary conditions

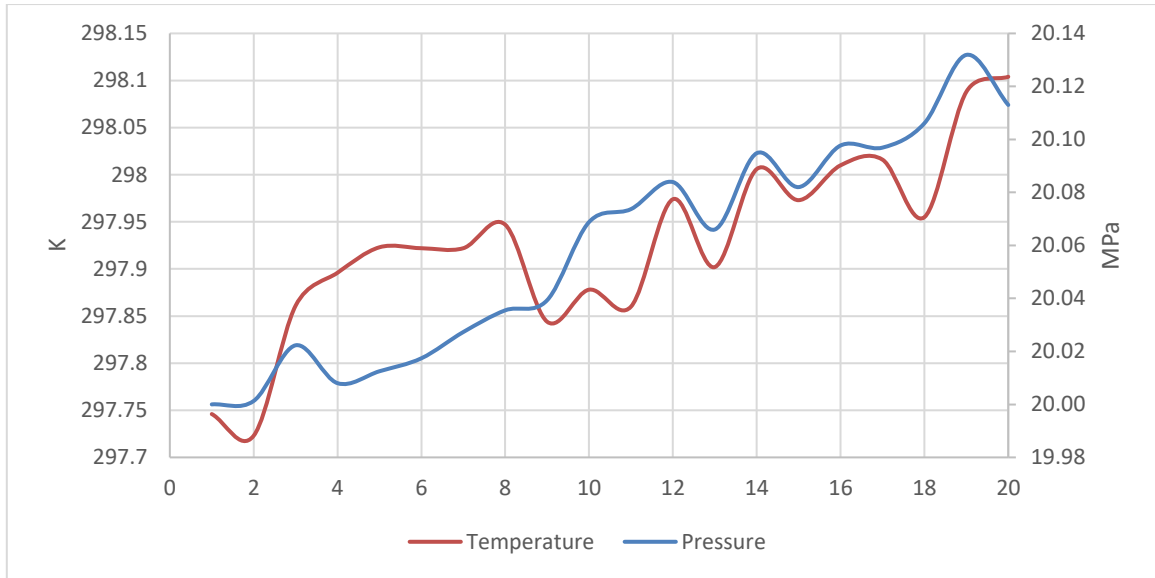| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 30 |
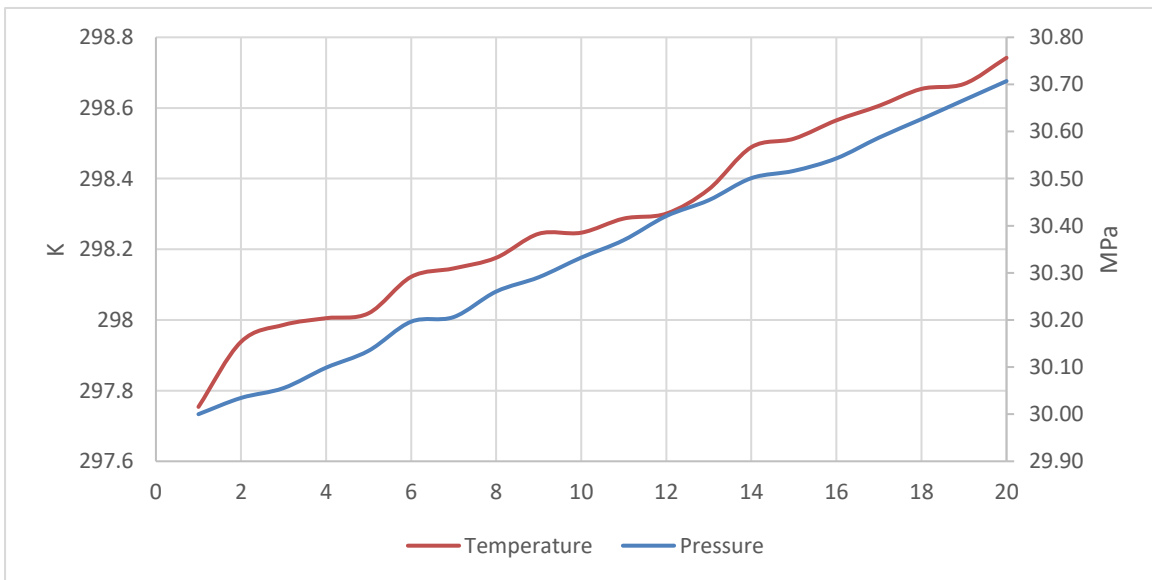


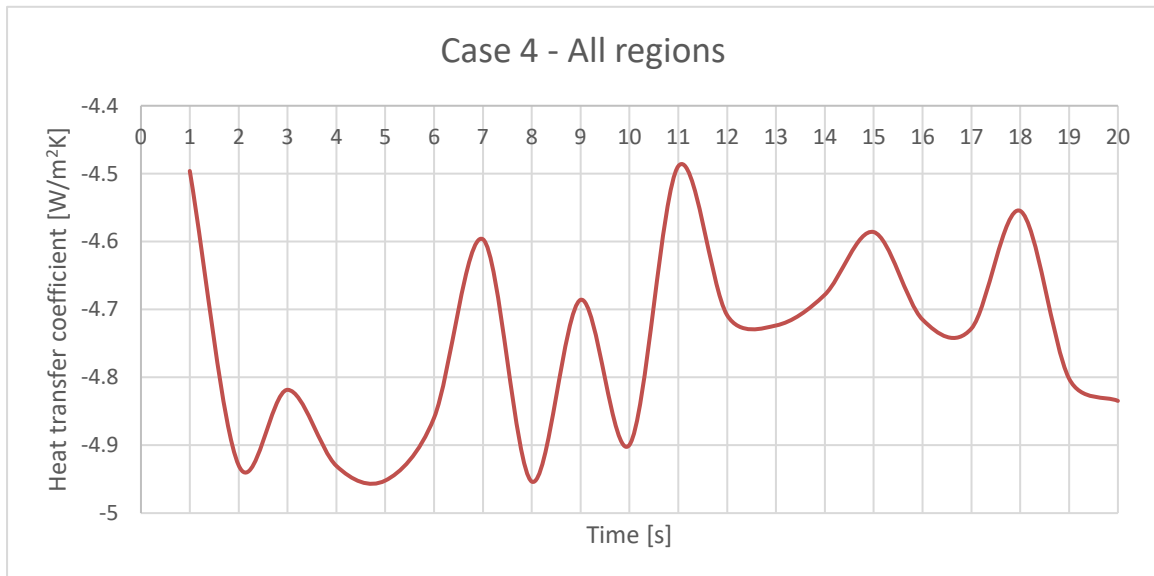Figure A-51 Case 11 average temperature and pressure
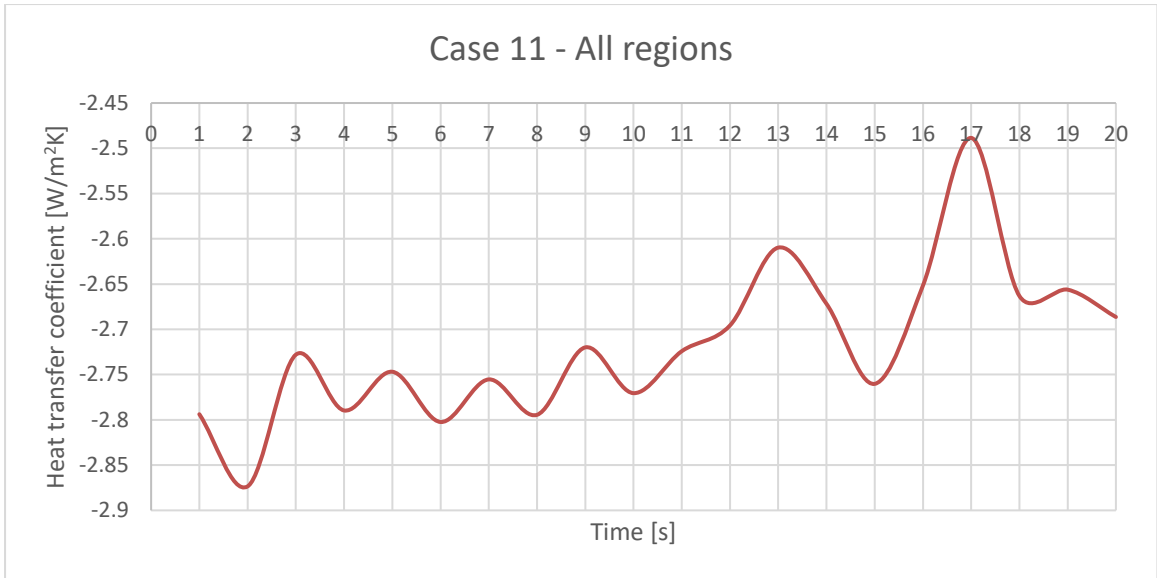
Figure A-52 Case 11 overall heat transfer coefficient for all regions
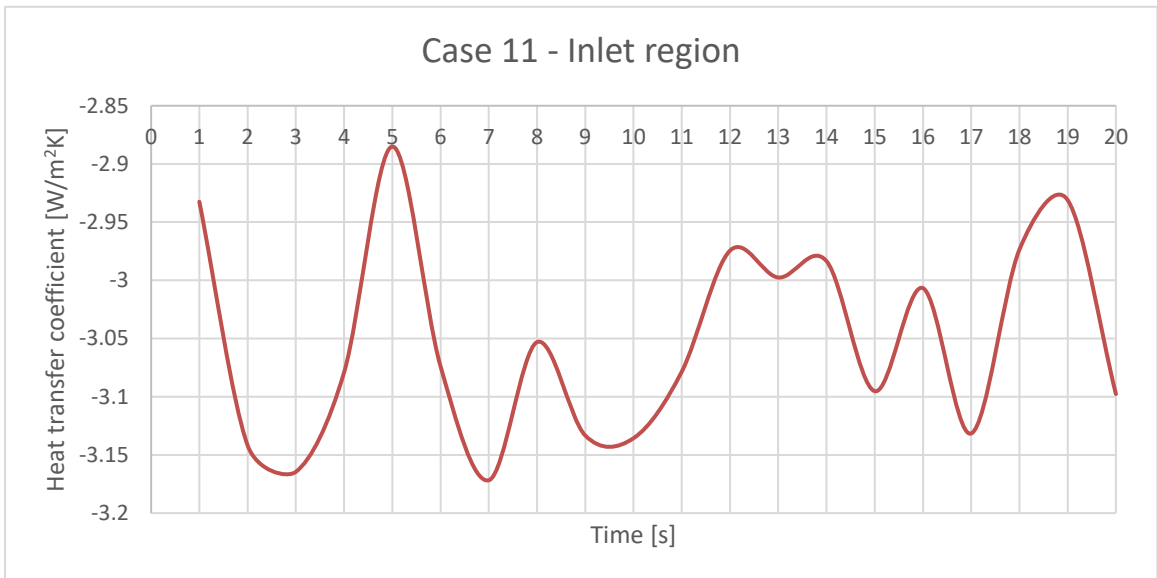


Figure A-53 Case 11 overall heat transfer coefficient for inlet region
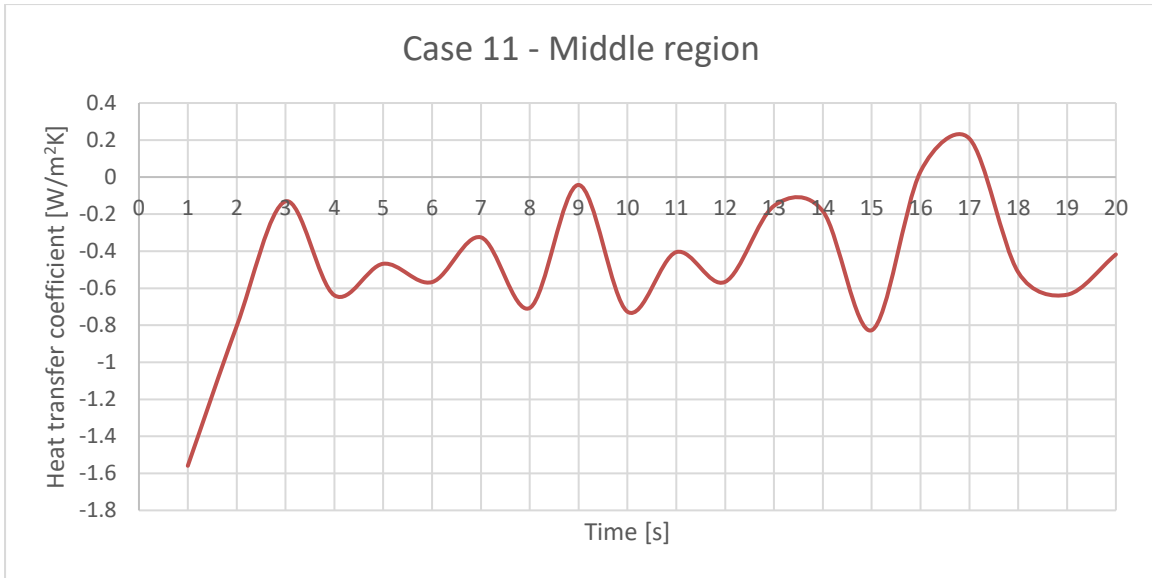
Figure A-54 Case 11 overall heat transfer coefficient for middle region
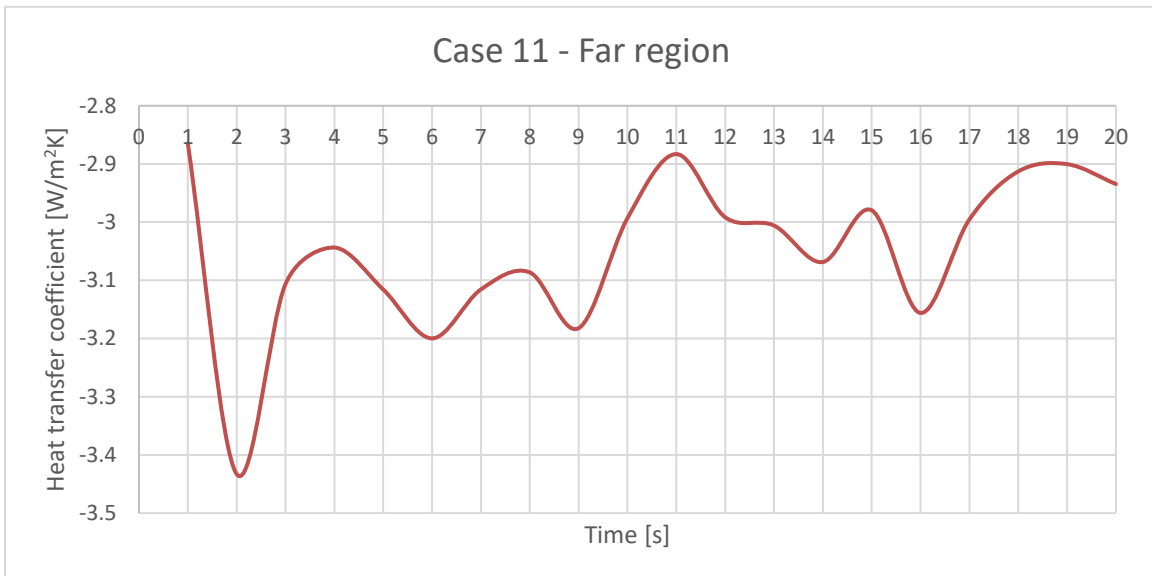


Figure A-55 Case 11 overall heat transfer coefficient for far region

## A.1.12. Case 12

Table A-12 Case 12 initial and boundary conditions

| Inlet Temperature [K] | Inlet mass flow [g/s] | Initial pressure [MPa] |
|---|---|---|
| 273.15 | 50 | 35 |
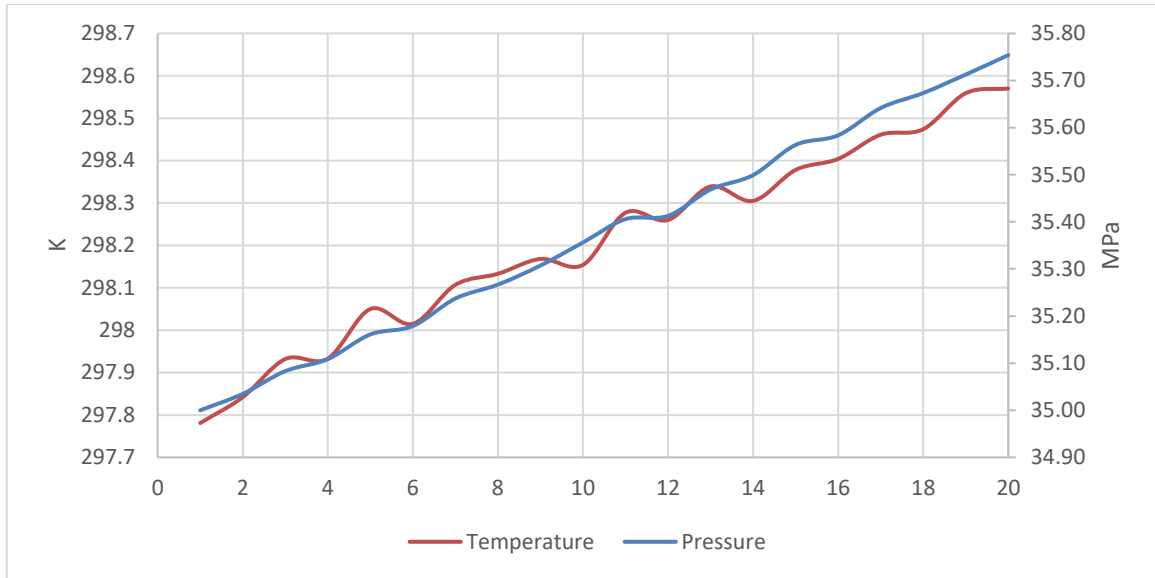
Figure A-56 Case 12 average temperature and pressure
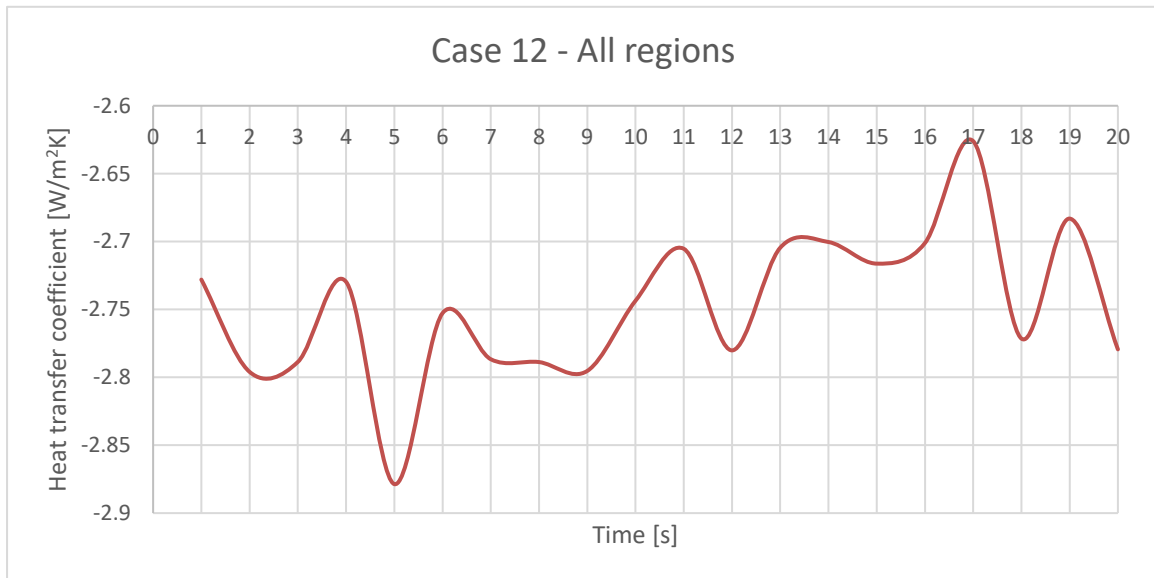


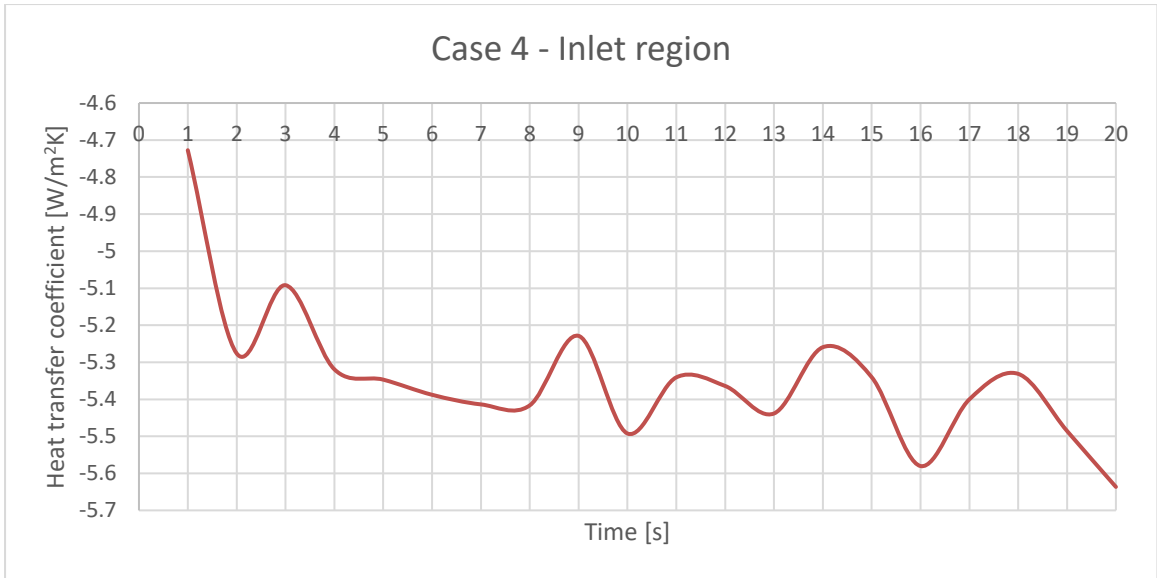Figure A-57 Case 12 overall heat transfer coefficient for all regions
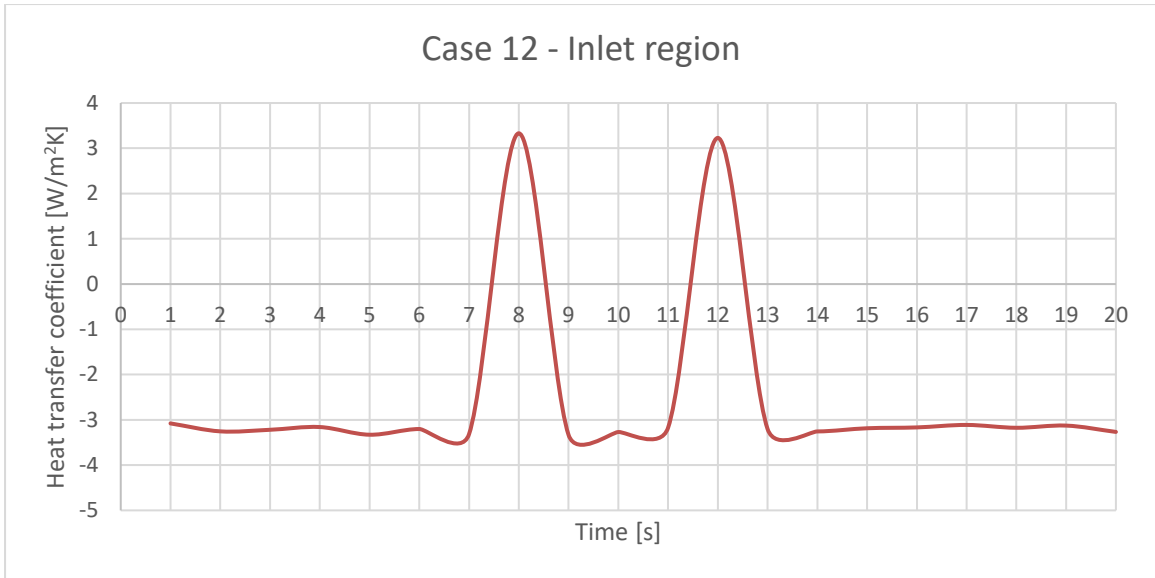
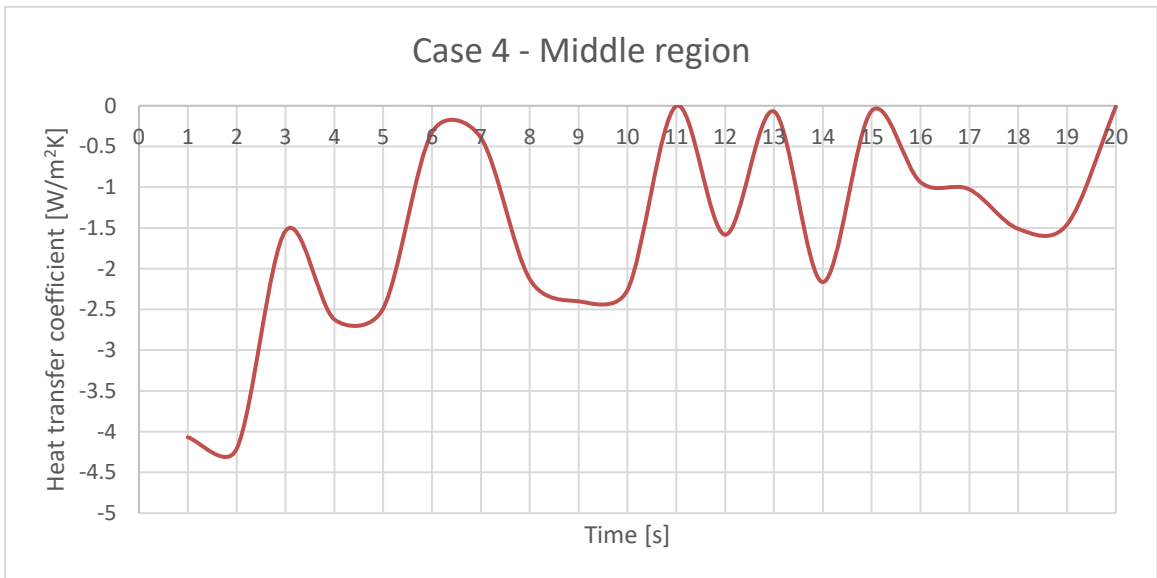Figure A-58 Case 12 overall heat transfer coefficient for inlet region



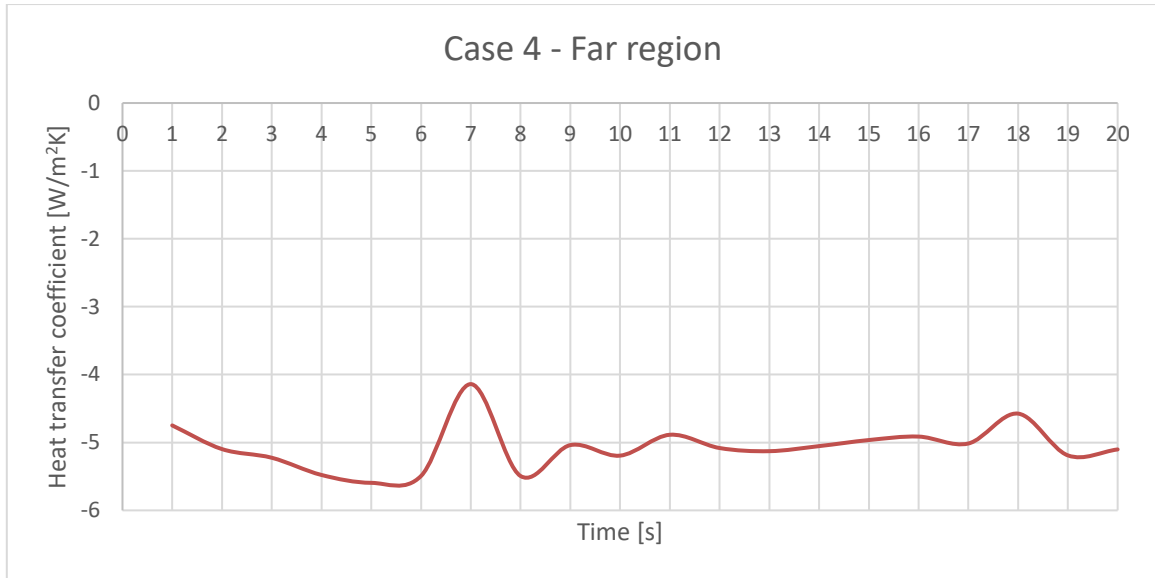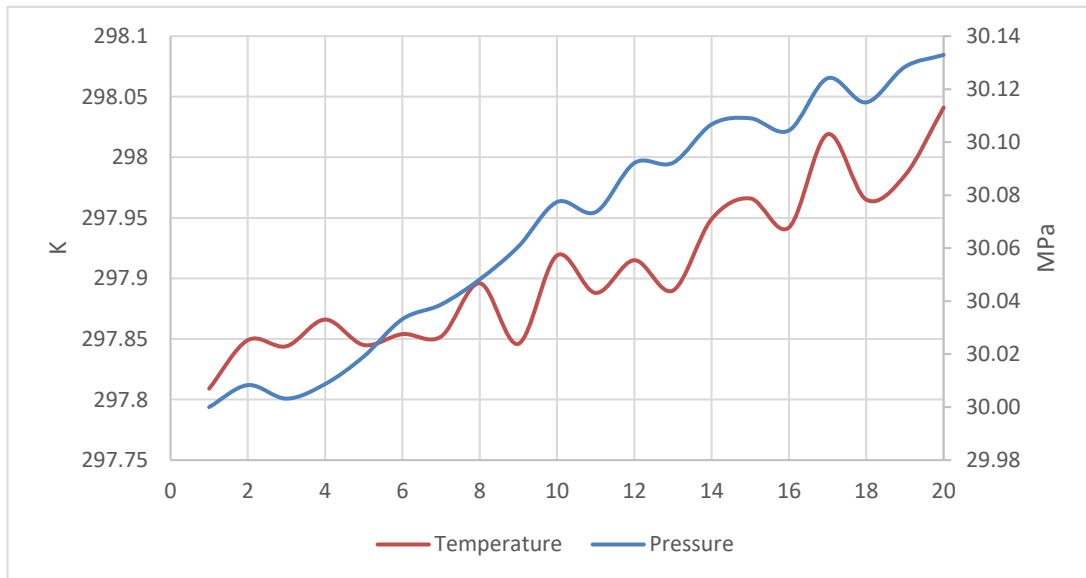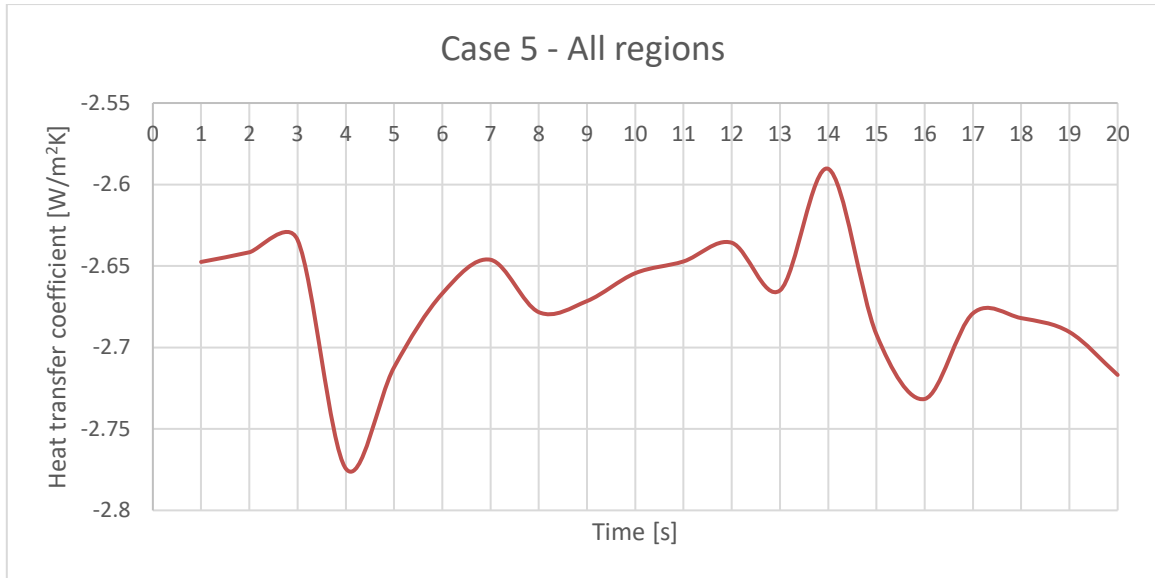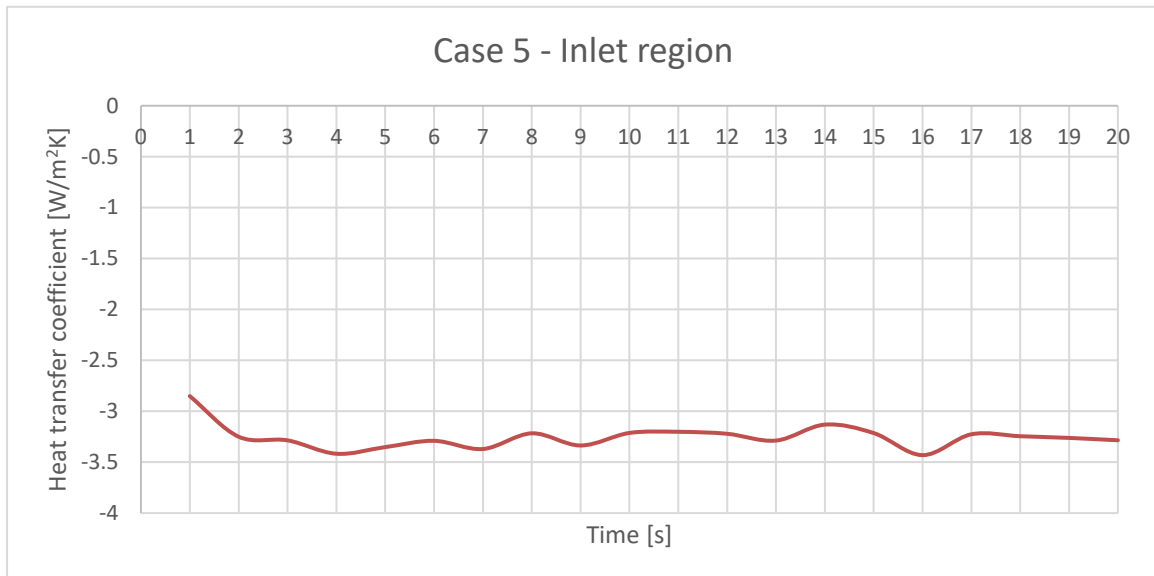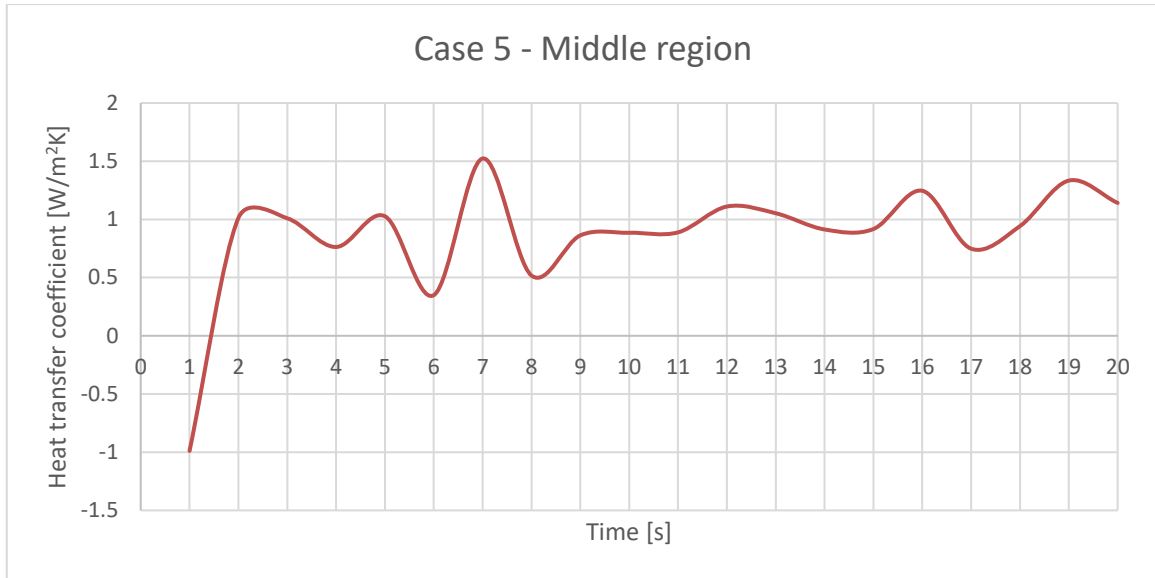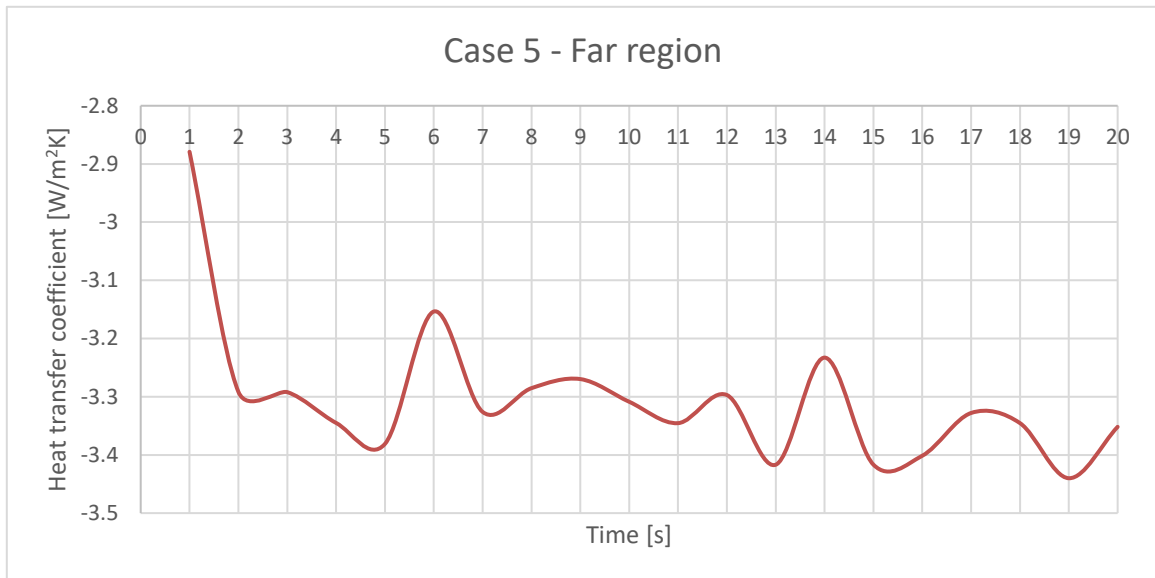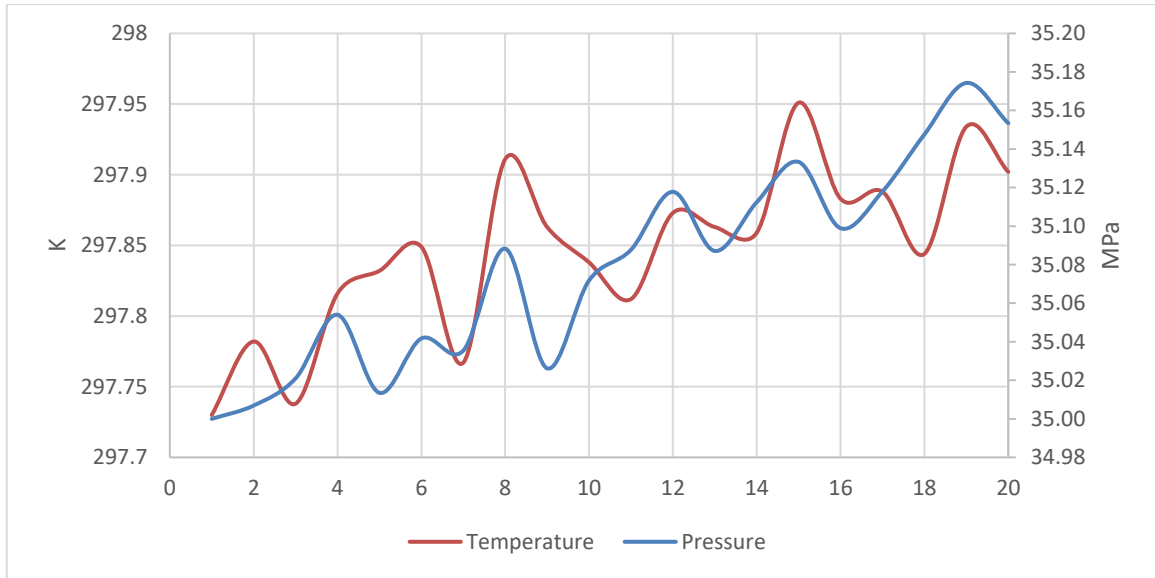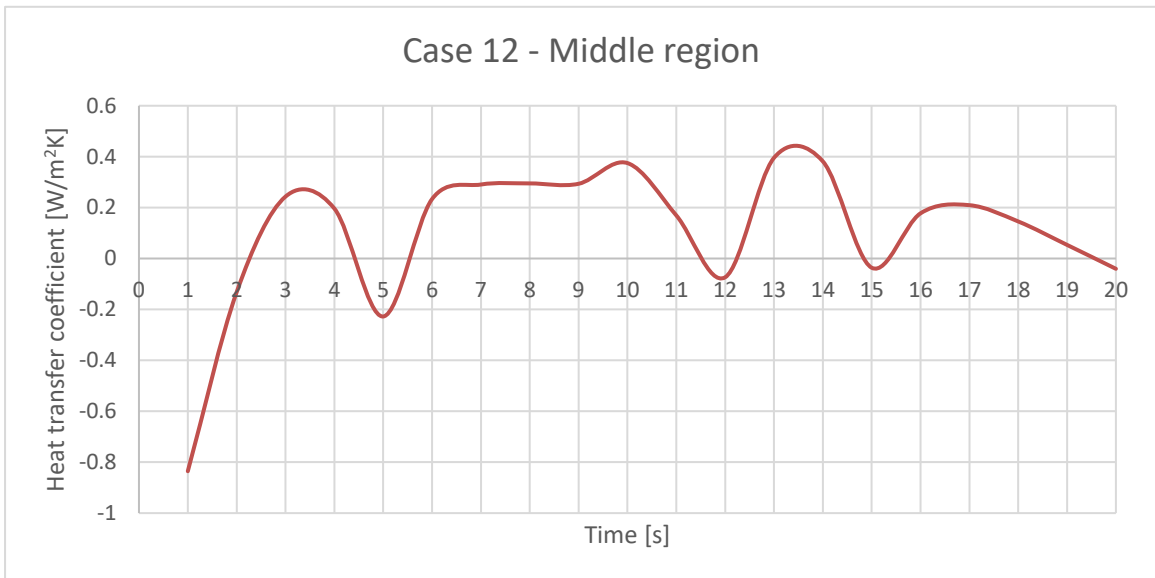Figure A-59 Case 12 overall heat transfer coefficient for middle region
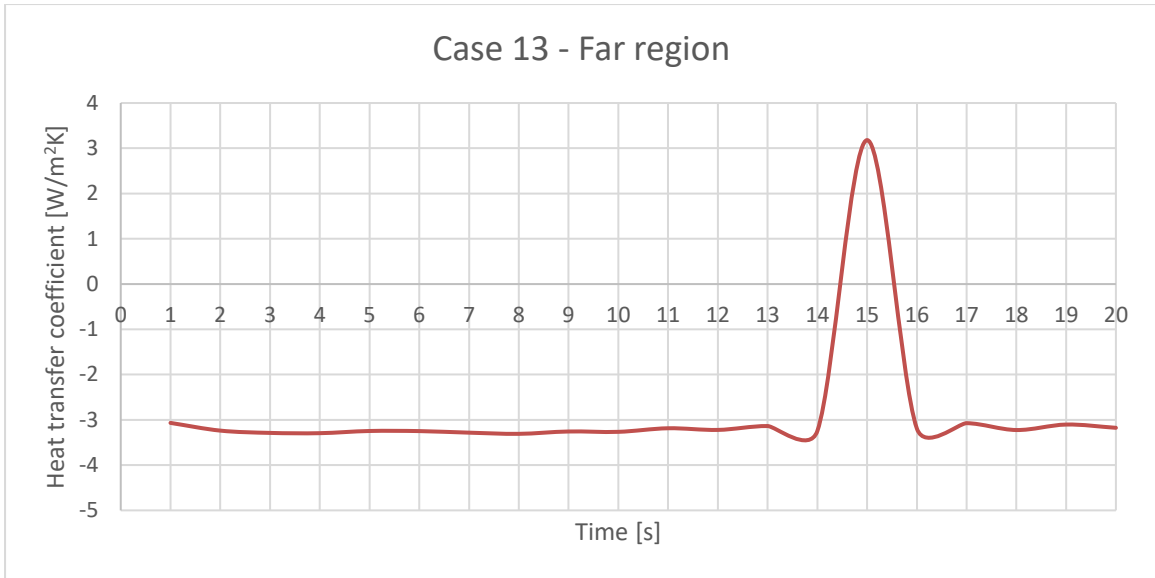
Figure A-60 Case 12 overall heat transfer coefficient for far region

# Appendix B. openFOAM dictionaries

In this appendix the openFOAM dictionaries data is shown. These dictionaries can be used to run the simulation and update it as needed. Only the dictionaries for Case 1 are shown. For the rest of the simulation cases, the dictionaries are similar, with only the change on the parameters as stated on Table 4.1 and Table 4.2for each of the cases.

## B.1.    0 folder

### B.1.1.    U

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [0 1 -1 0 0 0 0];
internalField   uniform (0 0 0);
boundaryField
{
    inlet
    {
        type            flowRateInletVelocity;
        massFlowRate    uniform 0.01;
        value           uniform (0 0 0);
    }
    inletupperWall
    {
        type            noSlip;
    }
    middleupperWall
    {
        type            noSlip;
    }
    farupperWall
    {
        type            noSlip;
    }
    inletlowerWall
    {
        type            noSlip;
    }
    middlelowerWall
    {
        type            noSlip;
```

```
    }
    farlowerWall
    {
        type            noSlip;
    }
    rightWall
    {
        type            noSlip;
    }
    leftWall
    {
        type            noSlip;
    }
     defaultFaces
    {
        type            empty;
    }
}
// ************************************************************** //
```

## B.1.2.      p

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions [1 -1 -2 0 0 0 0];
internalField   uniform 1e6;
boundaryField
{
    inlet
    {
        type            zeroGradient;

    }
    inletupperWall
    {
        type            zeroGradient;
    }
    middleupperWall
    {
        type            zeroGradient;
    }
    farupperWall
```

```
    {
        type            zeroGradient;
    }
    inletlowerWall
    {
        type            zeroGradient;
    }
    middlelowerWall
    {
        type            zeroGradient;
    }
    farlowerWall
    {
        type            zeroGradient;
    }
    rightWall
    {
        type            zeroGradient;
    }
    leftWall
    {
        type            zeroGradient;
    }
     defaultFaces
    {
        type            empty;
    }
}
// ************************************************************** //
```

### B.1.3.        T

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      T;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions      [0 0 0 1 0 0 0];
internalField   uniform 298.15;
boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform  273.15;
    }
```

```
inletupperWall
{
    type            externalWallHeatFluxTemperature;
    mode            coefficient;
    Ta              uniform 298.15;
    thicknessLayers (0.005);
    kappaLayers     (0.4);
    kappaMethod     fluidThermo;
    h               uniform 10;
    value           $internalField;  //initial temperature
    kappaMethod     fluidThermo;
}
middleupperWall
{
    type            externalWallHeatFluxTemperature;
    mode            coefficient;
    Ta              uniform 298.15;
    thicknessLayers (0.005);
    kappaLayers     (0.4);
    kappaMethod     fluidThermo;
    h               uniform 10;
    value           $internalField;  //initial temperature
    kappaMethod     fluidThermo;
}
farupperWall
{
    type            externalWallHeatFluxTemperature;
    mode            coefficient;
    Ta              uniform 298.15;
    thicknessLayers (0.005);
    kappaLayers     (0.4);
    kappaMethod     fluidThermo;
    h               uniform 10;
    value           $internalField;  //initial temperature
    kappaMethod     fluidThermo;
}
inletlowerWall
{
    type            externalWallHeatFluxTemperature;
    mode            coefficient;
    Ta              uniform 298.15;
    thicknessLayers (0.005);
    kappaLayers     (0.4);
    kappaMethod     fluidThermo;
    h               uniform 10;
    value           $internalField;  //initial temperature
    kappaMethod     fluidThermo;
}
middlelowerWall
{
    type            externalWallHeatFluxTemperature;
    mode            coefficient;
    Ta              uniform 298.15;
    thicknessLayers (0.005);
    kappaLayers     (0.4);
    kappaMethod     fluidThermo;
    h               uniform 10;
    value           $internalField;  //initial temperature
```

```
            kappaMethod      fluidThermo;
    }
    farlowerWall
    {
            type             externalWallHeatFluxTemperature;
            mode             coefficient;
            Ta               uniform 298.15;
            thicknessLayers  (0.005);
            kappaLayers      (0.4);
            kappaMethod      fluidThermo;
            h                uniform 10;
            value            $internalField;  //initial temperature
            kappaMethod      fluidThermo;
    }
    rightWall
    {
            type             externalWallHeatFluxTemperature;
            mode             coefficient;
            Ta               uniform 298.15;
            thicknessLayers  (0.005);
            kappaLayers      (0.4);
            kappaMethod      fluidThermo;
            h                uniform 10;
            value            $internalField;  //initial temperature
            kappaMethod      fluidThermo;
    }
    leftWall
    {
            type             externalWallHeatFluxTemperature;
            mode             coefficient;
            Ta               uniform 298.15;
            thicknessLayers  (0.005);
            kappaLayers      (0.4);
            kappaMethod      fluidThermo;
            h                uniform 10;
            value            $internalField;  //initial temperature
            kappaMethod      fluidThermo;
    }

    defaultFaces
    {
        type             empty;
    }
}
// ************************************************************* //
```

## B.1.4.      k

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------*/
```

```
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      k;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions      [0 2 -2 0 0 0 0];
internalField   uniform 0.0001;
boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform 0.007;
    }
    inletupperWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    middleupperWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    farupperWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    inletlowerWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    middlelowerWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    farlowerWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
     rightWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    leftWall
    {
        type            kqRWallFunction;
        value           uniform 0.0001;
    }
    #includeEtc "caseDicts/setConstraintTypes"
```

```
}
// ************************************************************* //
```

## B.1.5.        omega

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      omega;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
//omegaInlet     851.29;
dimensions      [0 0 -1 0 0 0 0];
internalField   uniform 0.0001;
boundaryField
{
    inlet
    {
        type            fixedValue;
        value           uniform 0.51290285;
    }
    inletupperWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    middleupperWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    farupperWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    inletlowerWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    middlelowerWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
```

```
    farlowerWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    rightWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    leftWall
    {
        type            omegaWallFunction;
        value           uniform 0.0001;
    }
    #includeEtc "caseDicts/setConstraintTypes"
}
// ************************************************************* //
```

## B.1.6.    nut

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      mut;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions      [0 2 -1 0 0 0 0];
internalField   uniform 0;
boundaryField
{
    inlet
    {
        type            calculated;
        value           uniform 0;
    }
    inletupperWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    middleupperWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
```

```
    farupperWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    inletlowerWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    middlelowerWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    farlowerWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    rightWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    leftWall
    {
        type            nutkWallFunction;
        value           uniform 0;
    }
    #includeEtc "caseDicts/setConstraintTypes"
}
// ************************************************************* //
```

### B.1.7.      alphat

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      alphat;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
dimensions      [1 -1 -1 0 0 0 0];
internalField   uniform 0;
boundaryField
{
    inlet
    {
```

```
        type            calculated;
        value           uniform 0;
    }
    inletupperWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    middleupperWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    farupperWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    inletlowerWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    middlelowerWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    farlowerWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    rightWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
    leftWall
    {
        type            compressible::alphatWallFunction;
        value           uniform 0;
    }
        #includeEtc "caseDicts/setConstraintTypes"
}
// ************************************************************* //
```

## B.2.    constant folder

### B.2.1.    thermophysicalProperties

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration      | Website:  https://openfoam.org
    \\  /    A nd            | Version:  7
     \\/     M anipulation   |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      thermophysicalProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

thermoType
{
    type            hePsiThermo;
    mixture         pureMixture;
    transport       polynomial;
    thermo          hPolynomial;
    equationOfState PengRobinsonGas;
    specie          specie;
    energy          sensibleEnthalpy;
}
mixture
{
    specie
    {
        molWeight       2.016;
    }
    equationOfState
    {
        Tc          32.994;
        Vc          65.5e-3;
        Pc          1293920.25;
        Zc          0.309;
        omega       -0.220;
    }
    thermodynamics
    {

        Hf  0;
        Sf  0;
        CpCoeffs<8> (6436.5105 63.161307 -0.1685728 1.5229265e-04 0 0 0
0);
    }
    transport
    {
        muCoeffs<8>     ( -0.135666 6.84115878e-02 -3.928747e-04
1.8996e-06 -5.23104e-09 7.4490972e-12 -4.250937e-15 0 );
        kappaCoeffs<8>  ( 2.009705e-02 3.234622e-04 2.1637249e-06 -
6.49151204e-09 5.52407932e-12 0 0 0 );
    }
}
// ************************************************************************* //
```

## B.2.2.　　turbulenceProperties

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield          | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration       | Website:  https://openfoam.org
    \\  /    A nd             | Version:  7
     \\/     M anipulation    |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "constant";
    object      turbulenceProperties;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
simulationType  RAS;
RAS
{
    RASModel        kOmegaSST;
    turbulence      on;
    printCoeffs     on;
}
// ********************************************************************* //
```

## B.3.　　system folder

## B.3.1.　　blockMeshDict

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield          | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration       | Website:  https://openfoam.org
    \\  /    A nd             | Version:  8
     \\/     M anipulation    |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    object      blockMeshDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

```
convertToMeters 1;
vertices
(
    (0 0 0) //0
    (2.2 0 0) //1
    (4.4 0 0) //2
    (6.6 0 0) //3
    (6.6 0.3 0) //4
    (6.6 0.4 0) //5
    (6.6 0.7 0) //6
    (4.4 0.7 0) //7
    (2.2 0.7 0) //8
    (0 0.7 0) //9
    (0 0.4 0) //10
    (0 0.3 0) //11
    (2.2 0.4 0) //12
    (2.2 0.3 0) //13
    (4.4 0.4 0) //14
    (4.4 0.3 0) //15

    (0 0 0.5497787144) //16
    (2.2 0 0.5497787144) //17
    (4.4 0 0.5497787144) //18
    (6.6 0 0.5497787144) //19
    (6.6 0.3 0.5497787144) //20
    (6.6 0.4 0.5497787144) //21
    (6.6 0.7 0.5497787144) //22
    (4.4 0.7 0.5497787144) //23
    (2.2 0.7 0.5497787144) //24
    (0 0.7 0.5497787144) //25
    (0 0.4 0.5497787144) //26
    (0 0.3 0.5497787144) //27
    (2.2 0.4 0.5497787144) //28
    (2.2 0.3 0.5497787144) //29
    (4.4 0.4 0.5497787144) //30
    (4.4 0.3 0.5497787144) //31
);
blocks
(
    hex (0 1 13 11 16 17 29 27) (150 25 1) simpleGrading (1 1 1)
    hex (11 13 12 10 27 29 28 26) (150 20 1) simpleGrading (1 1 1)
    hex (10 12 8 9 26 28 24 25) (150 25 1) simpleGrading (1 1 1)
    hex (1 2 15 13 17 18 31 29) (150 25 1) simpleGrading (1 1 1)
    hex (13 15 14 12 29 31 30 28) (150 20 1) simpleGrading (1 1 1)
    hex (12 14 7 8 28 30 23 24) (150 25 1) simpleGrading (1 1 1)
    hex (2 3 4 15 18 19 20 31) (150 25 1) simpleGrading (1 1 1)
    hex (15 4 5 14 31 20 21 30) (150 20 1) simpleGrading (1 1 1)
    hex (14 5 6 7 30 21 22 23) (150 25 1) simpleGrading (1 1 1)
);
edges
(
);
boundary
(
    inlet
    {
        type patch;
        faces
```

```
    (
        (11 10 26 27)
    );
}
leftWall
{
    type wall;
    faces
    (
        (0 11 27 16)
        (10 9 25 26)
    );
}
rightWall
{
    type wall;
    faces
    (
        (3 19 20 4)
        (4 20 21 5)
        (5 21 22 6)
    );
}

inletupperWall
{
    type wall;
    faces
    (
        (9 8 24 25)
    );
}

middleupperWall
{
    type wall;
    faces
    (
        (8 7 23 24)
    );
}
farupperWall
{
    type wall;
    faces
    (
        (7 6 22 23)
    );
}
inletlowerWall
{
    type wall;
    faces
    (
        (0 16 17 1)
    );
}
```

```
        middlelowerWall
        {
            type wall;
            faces
            (
                (1 17 18 2)
            );
        }
        farlowerWall
        {
            type wall;
            faces
            (
                (2 18 19 3)
            );
        }
);
mergePatchPairs
(
);
// ************************************************************* //
```

## B.3.2.  controlDict

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     | Website:  https://openfoam.org
    \\  /    A nd           | Version:  7
     \\/     M anipulation  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      controlDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * ** * * * * * * * * //
application     rhoCentralFoam;
startFrom       latestTime;
startTime       0;
stopAt          endTime;
endTime         20;
deltaT          0.005;
writeControl    adjustableRunTime;
writeInterval   1;
cycleWrite      0;
writeFormat     ascii;
writePrecision  10;
writeCompression off;
timeFormat      general;
timePrecision   6;
runTimeModifiable true;
adjustTimeStep  yes;
maxCo           0.5;
functions
```

```
{
    #includeFunc wallHeatFlux
    #includeFunc yPlus
inletMassFlow
 {
        type            surfaceFieldValue;
        functionObjectLibs ("libfieldFunctionObjects.so");
        enabled         true;
        writeControl    timeStep;
        writeInterval   500;
        log             true;
        writeFields     false;
        regionType      patch;
        name            inlet;
        operation       sum;
        fields
        (
            phi
        );
  }
probes
  {
            type            probes;
            libs            ("libsampling.so");
            writeControl    timeStep;
            writeInterval   500;

            fields
            (
                p
                T
            );

            probeLocations
            (
                (0.01 0.35 0)
                (6.59 0.35 0)
                (3.3 0.35 0)
                (3.3 0.05 0)
                (3.3 0.65 0)
            );
wallHeatFlux1
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("leftWall")
    writeControl    timeStep;
    writeInterval   1000;
}


wallHeatFlux2
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("rightWall")
    writeControl    timeStep;
    writeInterval   1000;
```

```
}

wallHeatFlux3
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("inletupperWall")
    writeControl    timeStep;
    writeInterval   1000;
}

wallHeatFlux4
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("middleupperWall")
    writeControl    timeStep;
    writeInterval   1000;
}


wallHeatFlux5
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("farupperWall")
    writeControl    timeStep;
    writeInterval   1000;
}

wallHeatFlux6
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("inletlowerWall")
    writeControl    timeStep;
    writeInterval   1000;
}


wallHeatFlux7
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("middlelowerWall")
    writeControl    timeStep;
    writeInterval   1000;
}

wallHeatFlux8
{
    type        wallHeatFlux;
    libs        ("libfieldFunctionObjects.so");
    patches     ("farlowerWall")
    writeControl    timeStep;
    writeInterval   1000;
}
```

```
}
// ************************************************************ //
```

## B.3.3.        fvSchemes

```
/*--------------------------------*- C++ -*----------------------------------*\
  =========                 |
  \\      /  F ield          | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration      | Website:  https://openfoam.org
    \\  /    A nd            | Version:  7
     \\/     M anipulation   |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
fluxScheme       Kurganov;
ddtSchemes
{
    default          Euler;
}
gradSchemes
{
    default          Gauss linear;
}
divSchemes
{
    default          none;
    div(phi,U) linearUpwind grad(U);
    div(phi,k)       Gauss upwind;
    div(phi,epsilon) Gauss upwind;
    div(phi,omega)  Gauss upwind;
    div(tauMC)       Gauss linear;
}
laplacianSchemes
{
    default          Gauss linear corrected;
}
interpolationSchemes
{
    default          linear;
    reconstruct(rho) vanLeer;
    reconstruct(U)   vanLeerV;
    reconstruct(T)   vanLeer;
}
snGradSchemes
{
    default          corrected;
}
wallDist
{
```

```
    method meshWave;
}
// ************************************************************* //
```

## B.3.4.    fvSolution

```
/*--------------------------------*- C++ -*----------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration      | Website:  https://openfoam.org
    \\  /    A nd            | Version:  7
     \\/     M anipulation   |
\*---------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
solvers
{
    "(rho|rhoU|rhoE)"
    {
        solver          diagonal;
    }
    "(k|epsilon|omega)"
    {
        solver          smoothSolver;
        smoother        symGaussSeidel;
        tolerance       1e-05;
        relTol          0.0;
    }
    "(k|epsilon|omega)Final"
    {
        $U;
        relTol          0.0;
    }
    U
    {
        solver          smoothSolver;
        smoother        GaussSeidel;
        nSweeps         2;
        tolerance       1e-09;
        relTol          0.0;
    }
    h
    {
        $U;
        relTol          0.0;
    }
}
// ************************************************************* //
```