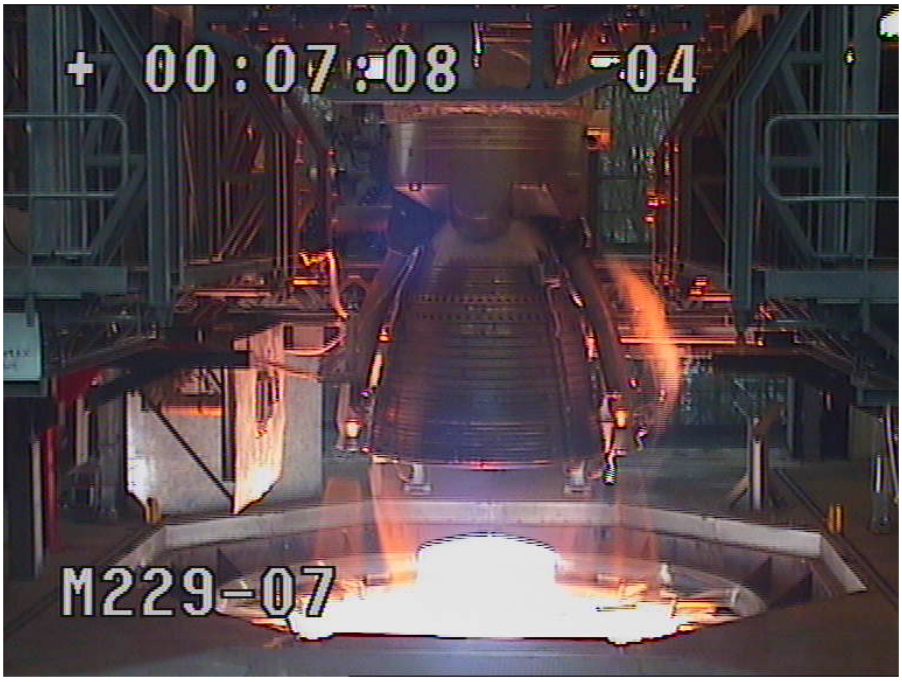
CHALMERS



CFD investigation of cooling flow in a rocket nozzle

A concept for one dimensional modeling with subdomains

Master's thesis in Solid and Fluid Mechanics

LILI HANSSON

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2012
Master's thesis 2012:37

MASTER'S THESIS IN SOLID AND FLUID MECHANICS

CFD investigation of cooling flow in a rocket nozzle

A concept for one dimensional modeling with subdomains

LILI HANSSON

Department of Applied Mechanics
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY

Gothenburg, Sweden 2012

CFD investigation of cooling flow in a rocket nozzle
A concept for one dimensional modeling with subdomains
LILI HANSSON

Master's thesis 2012:37
ISSN 1652-8557
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Sweden
Telephone: +46 (0)31-772 1000

Cover:
The Vulcain 2+ NE sandwich Demonstrator in full scale hot testing on Vulcain 2 engine M229, *Courtesy of Volvo Aero Corporation*

Chalmers reproservice
Gothenburg, Sweden 2012

CFD investigation of cooling flow in a rocket nozzle
A concept for one dimensional modeling with subdomains
Master's thesis in Solid and Fluid Mechanics
LILI HANSSON
Department of Applied Mechanics
Division of Fluid Dynamics
Chalmers University of Technology

Abstract

The main purpose of this work has been to study if it is possible to reduce the mesh resolution but still capture the *general* flow behavior and heat transfer in a rocket nozzle cooling channels by using a specific type of domain in computational fluid dynamics (CFD). The domain used in this work is a subdomain in the software *Ansys CFX* by the company *Ansys*. *Ansys CFX* is a software used to calculate fluid flow with the Navier-Stokes equations, so called CFD.

The geometry that the investigations was conducted on were a square tube with a length of 4.2 [m] connected with a inlet manifold and a outlet manifold and 4 other tubes. In order to verify the results a simulation were preformed in the beginning of the thesis that were compared with Volvo Aero's in-house code. The validation showed that, with the turbulence model SST $k-\omega$ and a friction loss coefficient, the temperature, pressure and velocity where matched with the simulation of the in-house code.

After verifying the boundary conditions and mesh, the second part of the thesis were to investigate how well *CFX* handles the interface between a very coarse mesh and a very fine mesh. These results showed that the interface affects the result but the disturbances are sufficiently low for the purpose of the task.

Finally the geometry with inlet manifold and outlet manifold and five tubes were simulated and evaluated in order to get a estimate of the time that it takes for the solution to converge. Comparing this simulation time with the time it takes for a fully resolved tube, the results showed a great improvement.

Keywords: CFD, rocket nozzle, nozzle, aerospace, turbulence, turbulence models, SST k-omega, real gas simulation, cooling, subdomain, friction loss coefficient

Preface

The 1 October 2012 GKN completed the aquisition of Volvo Aero Corporation and the company changed name from Volvo Aero to GKN Aerospace Engine Systems. This Master's thesis was carried out at Volvo Aero, department of Nozzles and Combustors, during 2012. The main purpose of the project was to investigate if it was possible to model, together with a subdomain, a one dimensional flow within a three dimensional flow using the commercial software *Ansys® CFX®* version 14.0. The results show that it is possible, with a very coarse grid, to capture the general flow behavior such as pressure -, temperature - and velocity variations. The results were compared with Volvo Aero in-house code *TCCOOL* which is based on experimental results and knowledge within the company. The reader is expected to have a background in fluid dynamics and Computational Fluid Dynamics (CFD).

Acknowledgements

I would like to thank my supervisors at Volvo Aero, Arne Boman and Dr. Jan Östlund, for an interesting and very knowledge demanding project. They have given me good support and have always answered all my questions. I also would like to thank my examiner at Chalmers, Professor Lars-Erik Eriksson, for his guidance, experience and knowledge about real gases and compressible flows. It would feel wrong not to mention and thank Tony Eriksson at EDR & Medeso's support for answering all my questions about *Ansys CFX*. Also I would like to thank my family for being so understanding and helping me to get through this period of my life.

Trollhättan, October 25, 2012
Lili Hansson

Contents

Nomenclature

σ_k	Constant for k-equation
$(\cdot)'$	The fluctuating part of the Reynold averaged flow property
$(\cdot)''$	The fluctuating part of the Favre averaged flow property
β^*	Model constant for the ε- equation, in k -ε-model $c_\mu = \beta^*$
ΔL	wall thickness
δ_{ij}	Kronecker delta
$\dot{Q}_c$	convective heat transfer rate per unit area for combustion gases
$\dot{Q}_L$	convective heat transfer rate per unit area for liquid cooling
$\dot{Q}_w$	convective heat transfer rate per unit area for the wall
$\frac{D}{Dt}$	Material derivative
λ	Thermal conductivity
λ_R	resistance coefficient of pipe flow
μ_t	Turbulent kinetic viscosity
ν	Kinematic viscosity
ω	Specific dissipation
$\overline{(\cdot)}$	Reynold time averaged part of flow property
Φ	Flow parameter
ρ	Density
σ_ω	Check more in lada
τ_{ij}	Viscous Stress tensor
ε_R	Equivalent sand grain roughness
$\widetilde{(\cdot)}$	Favre time averaged part of flow property
A_c	Cross section area
A_s	Surface area
C	Wetted perimeter
C_p	Heat conductivity at constant pressure
d_h	Hydraulic diameter
e_0	Total Energy
h	Static enthalpy
h_0	Total enthalpy
h_g	hot-gas film coefficient
h_L	liquid film coefficient
k	Turbulent kinetic energy
k_{loss}	Loss coefficient

k_w	thermal conductivity of the wall material
L	Length of the pipe
L	Nodal position in simulation
n	Number of channels
p	Pressure
Pr	Prantl number
Q	Total heat flux
q_j	Heat flux
QW	Heat flux in nodal position L
Re	Reynoldsnumber
S^*_{ij}	Trace-less viscous strain-rate
T_{aw}	Adiabatic wall temperature
T_L	liquid free stream temperature
T_{wc}	cold surface wall temperature
T_{wh}	hot side wall temperature
u_τ	Wall friction velocity
u_j	Instantaneous velocity
V_x	Velocity in pipe at point x
x_2	Distance from the wall, where x_2 is the direction normal to the wall
x_j	Cartesian Spatial direction
y^+	Normalized wall distance
z_x	Height position of point x

1 Introduction

When designing a rocket nozzle, one of the challenges is the complex physics and chemistry involved. In order to get a picture of the performance of the nozzle one uses several different methods. Consideration has to be taken to several different criteria. One very important criteria is that the material used can withstand the extremely high temperatures.

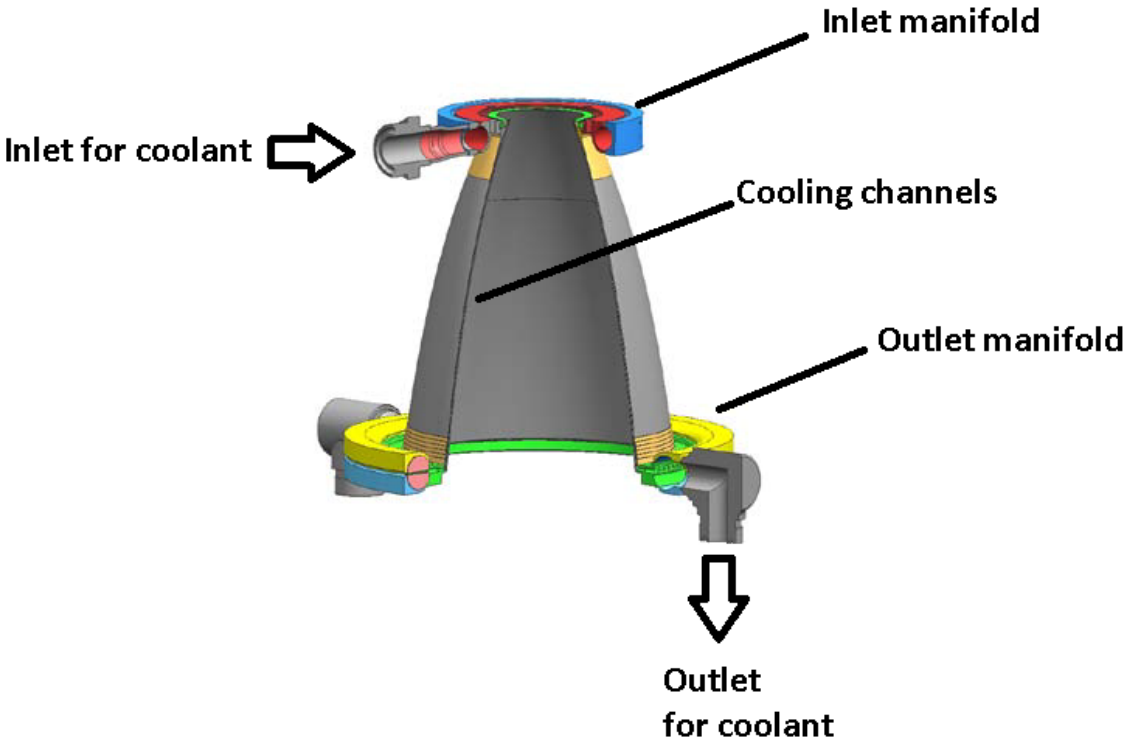


Figure 1.0.1 – *A CAD model of the subscale nozzle SCENE including the inlet- and outlet manifold (Picture reprinted from [5])*

In the core of the flame of the nozzle the temperature is around $T_0 = 3500K$ which makes cooling the material very important. Volvo Aero has for the last ten years developed and verified its patented "Sandwich" design and manufacturing methods for actively cooled nozzle extensions. This "Sandwich" manufacturing method allows for straight cooling channels following the nozzle shape axially. Volvo Aero is currently involved in ESA's[1] Future Launcher Preparatory Program where the staged combustion cycle High Thrust Engine (HTE) is in its early design phase. For this engine it is chosen to have a regenerative cooled Sandwich nozzle with a pressure level significantly higher than the Vulcain 2 nozzle [5]. This nozzle is called SCORE-D and to test the readiness of the technologies, Volvo Aero has designed the sub scale SCENE nozzle seen in figure 1.2.1. The nozzle is sub scaled since it is a smaller scale than the SCORE-D which is the full scale. Since it is regeneratively cooled it has a outlet manifold to pick up the coolant from the exit of the coolant channels.

To develop a nozzle Volvo Aero has a rigorous regulation for the design practice. The design has to go through several stages where the validity of the design and readiness of the technologies is tested. One method of testing the design is by CFD. Volvo Aero has developed, and uses, several one dimensional -, two dimensional

[1]ESA is short for European Space agency and they are responsible for drawing up European space programmes and carry it through. It consists of 19 member states and by coordinating the financial and intellectual resources of it member states it can undertake programs and activities far beyond the scope of a single European country. For more informations pleas see `http://www.esa.int`.

- and three dimensional in-house codes but also uses several three dimensional commercial softwares, to calculate the flow in the nozzle. It is desired to create a three dimensional 360°full simulation of the coolant flow with CFD at a early design phase, but this is not possible due to the complex and detailed geometry. The reader should be aware that the physics involved are very complex and this master's thesis will only look at one small part. By the coolant flow is ment where the coolant enters in to the nozzle, inlet manifold, then goes down through the Sandwich wall cooling channels, and exits from the outlet manifold, see figure 1.0.1.

1.1 Background

One of the challenges when working with CFD is the mesh resolution. In order to fully resolve the complex flow and solving the Navier-Stokes equations *(described further on in this thesis)* a very fine grid is needed. When using a fine grid, depending on the size of the domain, the amount of computer power needed to perform the calculations are too large for todays computers and the cost of performing such simulations is large. Therefore, models for example turbulence are used in order to be able to reduce mesh size. Often only a part of the whole model is simulated, depending on how much time that is dedicated to that specific simulation. For example, when a cooling channel is modeled about 10 million cells are needed. If instead the full cooling flow in the nozzle would be modeled and the nozzle had about 1000 channels the amount of cells needed to resolve this flow would be enormous and not even possible to perform. Therefore it is always a big challenge for simulation engineers to reduce grid resolution but increase the accuracy of the simulation. Volvo Aero has found, in other products and simulations, that the use of *subdomains* enables a coarser grid but still resolves the flow. A subdomain is simply a domain inside a larger domain.
Volvo Aero has a in-house developed code called TCCOOL. It is a one dimensional solver that computes the cooling performance of one cooling channel. It is a very accurate design tool but since it is a one-dimensional solver it is not able to capture massflow variations, therefore Volvo Aero is looking for a way to simulate the full nozzle in order to evaluate the performance of the cooling flow at an early design stage.

1.2 Purpose

As mentioned earlier, see chapter 1.1, Volvo Aero has found that subdomains might be able to reduce the grid resolution but still capture the general flow behavior. But it is unknown how low grid resolution that can be used before the information received from the subdomain is useless. Volvo Aero has developed a program that is validated against real test data, TCCOOL. It is possible to investigate if it is possible to use subdomains in the cooling channels by comparing subdomain modeling in CFX with this in house Volvo Aero code. The purpose of this master thesis is to investigate if it is possible to use subdomains in modeling the coolant flow in a rocket nozzle wall in order to reduce the mesh resolution significantly and finally to be able to model the full 360°cooling channels. Seen in figure 1.2.1 the variation of mass flow in the cooling channels creates a pattern of frost , at the top.

1.3 Problem description

If the full 360°coolant flow were to be simulated the number of elements needed to resolve the entire flow would exceed the given resources. Therefore it is of interest to investigate if it is possible to use subdomains in parts of the domain where the behavior of the flow is known, in order to decrease the mesh resolution. The problem will cover both simulations in the in-house code TCCOOL and CFD simulations, although focus is put on the

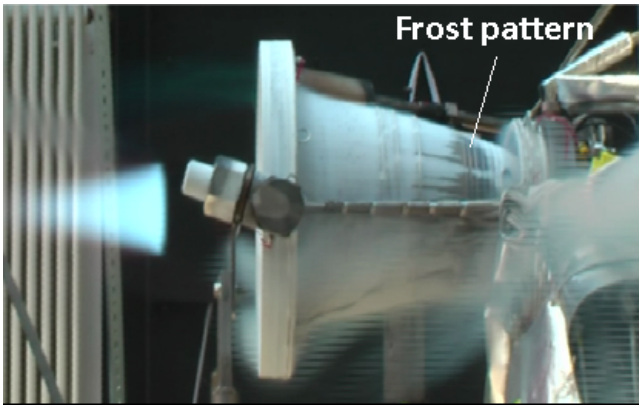


Figure 1.2.1 – *Frost pattern in test rig (Picture reprinted from [5])*

CFD simulations.
The work will be preformed in three main parts:

- Validation
- Interface between a coarse mesh and a finer mesh investigations
- Simulation with 5 channels

More on what the different parts mean will be explained in chapter 3. In order to receive the correct results from the CFD calculations the in-house code TCCOOL will be used to validate a channel flow with a very coarse mesh together with a subdomain.

1.4 Method

The thesis work started with a literature study where some papers about nozzles and the general flow behavior were studied. Also some literature about subdomains were studied but since very few investigations are made with the subdomains used the way that Volvo Aero intends there were not any papers found on this subject. The next task was to investigate subdomains and also something called porous domain, explained in chapter 2.6. There are mainly three softwares used during this thesis. To run the computational fluid dynamics (CFD), the commercial software *ANSYS ® CFX ® 14.0* were chosen because this is the commercial CFD program used to simulate fluid flow when a nozzle is developed at Volvo Aero. To create the mesh for the CFD simulations the commercial software *ANSYS® ICEM CFD™ 14.0* were used. The in-house code *TCCOOL* is a one-dimensional aero-thermal code. The code gives the performance of the cooling channels. Further information about these softwares can be found under chapter 3.

2 Theory

In this chapter the theory involved in this work is presented. The theory for the friction factor used in the subdomain is presented. The governing equations which are the equations that are solved in the CFD simulations, the turbulence model used to close the governing equations. Some simple heat transfer will be presented and how the equation of state for the real gas is calculated. Finally how the subdomain will affect the flow is presented.

2.1 Turbulent flow in non-circular pipes

The flow in a non-circular pipe have been investigated. From the Bernoulli equation for *unsteady frictionless flow along a streamline* , equation 2.1.2 , the simplified form for incompressible steady flow for pressure loss between section 1 and 2 of constant cross section area defined for friction and velocity distribution in a turbulent flow in a non-circular pipe, with hydraulic diameter d_h and the length over where the pressure loss occurs L, as [23]:

$$\frac{p_1 - p_2}{L} = \frac{1}{2}\frac{\lambda_R}{d_h}\rho\overline{u}^2, \qquad d_h = \frac{4A_c}{C} \tag{2.1.1}$$

Where the Bernoulli equation is defined as [27]:

$$\int_1^2 \frac{\partial V}{\partial t}\,\mathrm{d}s + \int_1^2 \frac{\partial p}{\rho}\,\mathrm{d}s + \frac{1}{2}\left(V_2^2 - V_1^2\right) + g\left(z_2 - z_1\right) = 0 \tag{2.1.2}$$

Where the dynamic pressure is defined as:

$$P_{dyn} \equiv \frac{1}{2}\rho\overline{u}^2 \tag{2.1.3}$$

The pressure loss over the pipe then depend on, with some rearrangement of equation 2.1.1:

$$\frac{p_1 - p_2}{LP_{dyn}} = \frac{\lambda_R}{d_h} \tag{2.1.4}$$

And from equation 2.1.4 the coefficient of loss, k_{loss}, can be defined:

$$k_{loss} \equiv \frac{\lambda_R}{d_h} \tag{2.1.5}$$

The resistance coefficient of pipe flow can be found in a Moody chart. This is a very simple way to find the value for the friction factor once, but since in this case, it is needed to be found a new value in every iteration and in every computational cell, the Moody chart becomes very laborious to program in the computer code. Therefore the loss coefficient is solved using the Serghides friction factor equations [24] with the coefficients A, B and C:

$$A = -2\log_{10}\frac{\frac{\varepsilon_R}{d_h}}{3.7} + \frac{12}{Re} \tag{2.1.6}$$

$$B = -2\log_{10}\frac{\frac{\varepsilon_R}{d_h}}{3.7} + \frac{2.51A}{Re} \tag{2.1.7}$$

$$C = -2\log_{10}\frac{\frac{\varepsilon_R}{d_h}}{3.7} + \frac{2.51B}{Re} \tag{2.1.8}$$

The resistance coefficient of pipe flow is calculated by:

$$\lambda_R = \left(A - \frac{(B-A)^2}{C-2B+A}\right) \tag{2.1.9}$$

The Reynolds number, Re, is defined by:

$$Re = \frac{\rho u d_h}{\mu} \tag{2.1.10}$$

Where k_{loss} can be seen as a function depending on Reynolds number, Re, and *equivalent sand grain roughness* of the pipe, ε_R.

2.2 Governing Equations

When dealing with compressible flow it is important to define all governing equations in the instantaneous form, chapter 2.2.1. Instead of letting the density be constant and simplifying the equations for incompressible flow, a Favre averaging is used, chapter 2.2.2 for compressible flow. This process will be explained in the following chapters.

2.2.1 Instantaneous Equations

The governing equations, the continuity equation 2.2.1, momentum equation 2.2.2, and energy equations 2.2.3, is described in the instantaneous velocity field:

$$\frac{d\rho}{dt} + \frac{\partial}{\partial x_j}(\rho u_j) = 0 \tag{2.2.1}$$

$$\frac{d}{dt}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij} - \tau_{ij}) = 0 \tag{2.2.2}$$

$$\frac{d}{dt}(\rho e_0) + \frac{\partial}{\partial x_j}(\rho u_j e_o + u_j p + q_j - u_i \tau_{ij}) = 0 \tag{2.2.3}$$

The total energy, e_o and total enthalpy, h_0 is defined as:

$$e_0 \equiv e + \frac{u_k u_k}{2}, \qquad h_0 \equiv h + \frac{u_k u_k}{2} \tag{2.2.4}$$

Enthalpy depend on internal energy from the following relation [1]:

$$h \equiv e + p\nu \tag{2.2.5}$$

Where e is the internal energy, p is pressure and ν is viscosity. The first and second term in the space derivative in the energy equation 2.2.3 can be written, using equation 2.2.4 and 2.2.5:

$$\rho u_j e_o + u_j p \equiv \rho u_j \left(h + \frac{u_k u_k}{2}\right) \tag{2.2.6}$$

So the final form of the instantaneous governing equations are:

$$\frac{d\rho}{dt} + \frac{\partial}{\partial x_j}(\rho u_j) = 0 \tag{2.2.7}$$

$$\frac{d}{dt}(\rho u_i) + \frac{\partial}{\partial x_j}(\rho u_i u_j + p\delta_{ij} - \tau_{ij}) = 0 \tag{2.2.8}$$

$$\frac{d}{dt}(\rho e_0) + \frac{\partial}{\partial x_j}(\rho u_j h_o + q_j - u_i \tau_{ij}) = 0 \tag{2.2.9}$$

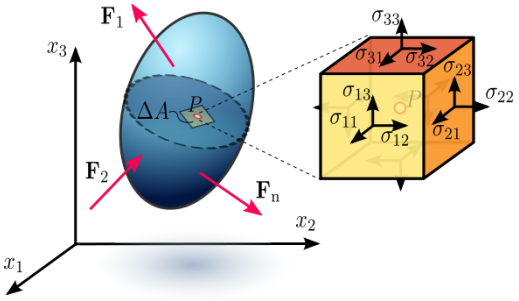


Figure 2.2.1 – *Stress in a fluid element in continuum*

The governing equations are not closed. In order to close these equations it is necessary to define the equation of state. In this case the gas is considered as a real gas, see chapter 2.5. Where the viscous stress tensor can be defined for a Newtonian fluid as:

$$\tau_{ij} = 2\mu S^*_{ij} \tag{2.2.10}$$

The trace-less, i.e when taking the trace of the tensor it equals zero, viscous-stress tensor is used to separate the thermodynamic pressure, p, with the strain-rate tensor S_{ij}. It can be seen from fig 2.2.1 that the normal stresses acting upon an fluid element can be interpret as a pressure. Therefore it is convenient to, in compressible flow, divide the stress tensor, τ_{ij}, in thermodynamic pressure p and the trace-less viscous strain-rate tensor, S^*_{ij} defined by [4]:

$$S^*_{ij} \equiv \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{1}{3}\frac{\partial u_k}{\partial x_k}\delta_{ij} \tag{2.2.11}$$

Fourier's law states that the heat-flux, q_j, depend on the temperature gradient and a thermal conductivity, λ:

$$q_j = -\lambda\frac{\partial T}{\partial x_j} \tag{2.2.12}$$

The thermal conductivity is defined as [25] :

$$\lambda \equiv C_p\frac{\mu}{Pr} \tag{2.2.13}$$

The Prandtl number is the kinetic viscosity divided by the thermal diffusivity. It shows the relative importance of viscosity and heat conduction in the flow field. It is a convenient value to use when both the kinematic viscosity and the density is varying since it is common that the Prandtl number is close to constant. In this case, the thermal conductivity is defined, see chapter 3.1, so hence the Prandtl number is not used in this case.

The Prandtl number, Pr, is defined [25]:

$$Pr \equiv \frac{\nu}{\frac{\lambda}{C_p \rho}} \tag{2.2.14}$$

2.2.2 Favre Averaged Equations

In order to solve the instantaneous equations explained in the previous section the equations are divided in one mean part and one fluctuating part. In compressible flow Favre averaging is used to separate the turbulent fluctuations from the mean-flow instead of the classic Reynolds averaging. Since the gas used is a real gas and a heat flux is applied creating a temperature increase, the flow becomes compressible and therefore the Favre averaging is necessary.

Averaging

In Reynolds averaging [7] the fluid flow parameter Φ is divided in a time-averaged part $\overline{\Phi}$ and a fluctuating part Φ'.

$$\Phi(t) = \overline{\Phi} + \Phi'(t), \quad \overline{\Phi} = \lim_{T \to \infty} \frac{1}{T} \int_0^T \Phi(t) dt \tag{2.2.15}$$

When Favre averaging [9] the flow parameter is instead divided in a time averaged density weighted part $\widetilde{\Phi}$ and fluctuating part Φ''.

$$\Phi(t) = \widetilde{\Phi} + \Phi''(t), \quad \widetilde{\Phi} = \frac{1}{\overline{\rho}} \lim_{T \to \infty} \frac{1}{T} \int_0^T \rho \Phi(t) dt \tag{2.2.16}$$

Note that for the definitions above $\overline{\Phi'} = 0$, but $\overline{\Phi''} \neq 0$.
No mass flux occurs over a Favre-average streamline and this makes the Favre-averaged variables in the governing equations more compact, this would not be possible by the Reynolds averaging.

Open Turbulent equations

In order to solve the instantaneous continuity equation 2.2.7, - momentum equation 2.2.8 and - energy equation 2.2.9 are time averaged. For the compressible flow the density weighted averaging decomposition is used, see chapter 2.2.2, for u_i and e_0, and the Reynolds time average decomposition 2.2.15 for ρ and p which gives the exact open turbulent governing equations:

$$\frac{d\overline{\rho}}{dt} + \frac{\partial}{\partial x_i} (\overline{\rho} \widetilde{u_j}) = 0 \tag{2.2.17}$$

$$\frac{\partial}{\partial t}\left(\overline{\rho}\widetilde{u_i}\right) + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_i}\widetilde{u_j} + \overline{p}\delta_{ij} + \overline{\rho u_i'' u_j''} - \overline{\tau_{ij}}\right] = 0 \tag{2.2.18}$$

$$\frac{\partial}{\partial t}\left(\overline{\rho}\widetilde{e_0}\right) + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j}\widetilde{h_0} + \underbrace{\overline{\rho u_j'' h_0''}}_{(\mathrm{A}^*)} + \overline{q_j} - \overline{u_i \tau_{ij}}\right] = 0 \tag{2.2.19}$$

Where the density averaged total energy $\widetilde{e_0}$ is defined as:

$$\widetilde{e_0} \equiv \widetilde{e} + \frac{\widetilde{u_k}\widetilde{u_k}}{2} + k \tag{2.2.20}$$

Where k is the turbulent kinetic energy and defined as:

$$k \equiv \frac{1}{2}\widetilde{u_k'' u_k''} \tag{2.2.21}$$

Note that $\widetilde{u_k'' u_k''}$ is the trace of $\widetilde{u_i'' u_j''}$ which is defined as:

$$\widetilde{u_k'' u_k''} \equiv \widetilde{u_1'' u_1''} + \widetilde{u_2'' u_2''} + \widetilde{u_3'' u_3''} \tag{2.2.22}$$

Equation 2.2.17, 2.2.18 and 2.2.19 are the Favre averaged Navier-Stokes equations and when solving them the variables of interest are $\overline{\rho}$, $\widetilde{u_i}$ and $\widetilde{e_0}$.

Approximations

Some simplifications can be done to the exact open turbulent equations, equation 2.2.17, 2.2.18 and 2.2.19:

$$\overline{\tau_{ij}} = \widetilde{\tau_{ij}} + \overline{\tau_{ij}''} \tag{2.2.23}$$

$$\overline{q_j} = \overline{\lambda \frac{\partial T}{\partial x_j}} = -\lambda \frac{\partial \widetilde{T}}{\partial x_j} - \lambda \frac{\partial \overline{T''}}{\partial x_j} \tag{2.2.24}$$

$$\overline{u_i \tau_{ij}} = \widetilde{u_i}\widetilde{\tau_{ij}} + \overline{u_i'' \tau_{ij}} + \widetilde{u_i}\overline{\tau_{ij}''} \tag{2.2.25}$$

Note that the Fouriers law and real gas relations have been applied. Also the fluctuations in the molecular viscosity, μ, have been neglected [14]. In order to close the energy equation it is needed to assume the perfect gas law. It is possible to assume the perfect gas law at this stage since the enthalpy depend on the temperature. In reality the gas *is* real gas and the enthalpy h depend on both temperature and pressure. But the pressure dependence is small as can be seen in figure 2.2.3 and figure 2.2.2. Therefore term (A*) is rephrased with the

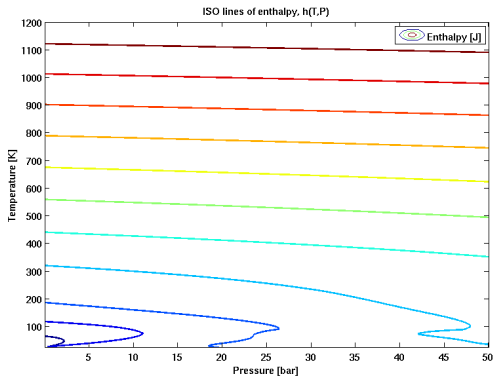


Figure 2.2.2 – *Contours of constant enthalpy in pressure and temperature*

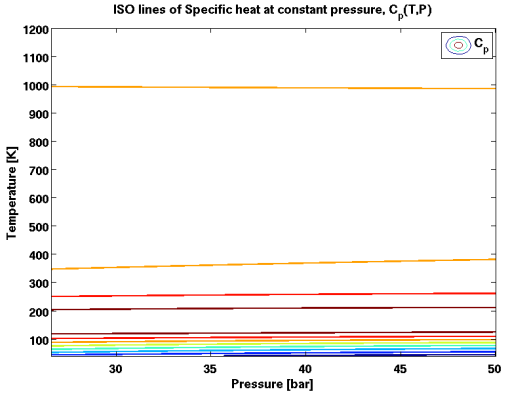


Figure 2.2.3 – *Contours of constant specific heat at constant pressure in pressure and temperature*

perfect gas law, $h = C_p T$:

$$\overline{\rho u_j'' h_0''} = C_p \overline{\rho u_j'' T''} + \widetilde{u_i} \overline{\rho u_i'' u_j''} + \overline{\rho \frac{1}{2} u_i'' u_i''} \tag{2.2.26}$$

Inserting these simplifications into 2.2.17, 2.2.18, 2.2.19 give:

$$\frac{d\rho}{dt} + \frac{\partial}{\partial x_j}\left(\rho \widetilde{u_j}\right) = 0 \tag{2.2.27}$$

$$\frac{d}{dt}\left(\overline{\rho}\widetilde{u_i}\right) + \frac{\partial}{\partial x_j}\left[\rho \widetilde{u_i}\widetilde{u_j} + \overline{p}\delta_{ij} + \underbrace{\overline{\rho u_i'' u_j''}}_{(1^*)} - \widetilde{\tau_{ij}} - \underbrace{\overline{\tau_{ij}''}}_{(2^*)}\right] = 0 \tag{2.2.28}$$

$$\frac{d}{dt}\left(\overline{\rho}\widetilde{e_0}\right) + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j}\widetilde{h_0} + \underbrace{C_p \overline{\rho u_j'' T''}}_{(3^*)} + \underbrace{\widetilde{u_i} \overline{\rho u_i'' u_j''}}_{(4^*)} + \underbrace{\frac{\overline{\rho u_j'' u_i'' u_i''}}{2}}_{(5^*)} - \underbrace{\lambda \frac{\partial \widetilde{T}}{\partial x_j}}_{(6^*)} - \underbrace{\lambda \frac{\partial \overline{T''}}{\partial x_j}}_{(7^*)} - \widetilde{u_i}\widetilde{\tau_{ij}} - \underbrace{\overline{u_i'' \tau_{ij}}}_{(8^*)} - \underbrace{\widetilde{u_i}\overline{\tau_{ij}''}}_{(9^*)}\right] = 0 \tag{2.2.29}$$

Terms from (1*) to (9*) are unknowns and will need models in order to close the governing equations, the assumptions and the models will be explained in the following chapter.

2.3 Turbulence Model

The turbulent flow regime is characterized by irregularity, diffusivity, large Reynolds numbers, three-dimensionality, dissipation and continuum [7]. Turbulent flow is irregular, random and chaotic. The flow consists of a spectrum of different scales. Even though the turbulence is chaotic it is still deterministic and can therefore still be solved by the Navier-Stokes equations, equations 2.2.1, 2.2.2 and 2.2.3. But when trying to solve these equations for turbulent flow it requires a very high mesh resolution to capture the smallest scales. Therefore several different turbulent models has been developed. To be able to use a coarser mesh but still capture the effects of the smallest scales. Following chapters will describe the turbulence model used in this thesis.

The Boussinesq assumption

The unknown Reynold stresses, term (1*) and (4*) are modeled with the Boussinesq assumption. The assumptions introduce an eddy viscosity, ν_t, which replaces six unknown turbulent stresses with one unknown.

$$\tau_{ij}^{turb} \equiv -\overline{\rho u_i'' u_j''} = 2\nu_t \widetilde{S_{ij}^*} + \frac{2}{3}\overline{\rho}\tilde{k}\delta_{ij} \tag{2.3.1}$$

The turbulent viscosity will also be estimated with a turbulence model. Term (2*) and (9*) can be neglected when:

$$|\widetilde{\tau_{ij}}| >> |\overline{\tau_{ij}''}| \tag{2.3.2}$$

Term (3*) can be modeled using a gradient approximation:

$$q_j^{turb} \equiv C_p \overline{\rho u_j'' T''} = -C_p \frac{\nu_t}{Pr_t} \frac{\partial \overline{T}}{\partial x_j} \tag{2.3.3}$$

Terms (5*) and (8*) can be neglected if they are small compared to the turbulent kinetic energy [14]. For most flows below the hyper-sonic regime this assumption is valid and is therefore also applicable for this case. Term (7*) stems from the Favre averaging and can be neglected because:

$$\left| \frac{\partial^2 \widetilde{T}}{\partial x_j^2} \right| >> \left| \frac{\partial^2 \overline{T''}}{\partial x_j^2} \right| \tag{2.3.4}$$

Summarizing the governing equation with the assumptions made above yields the open turbulent governing equations [14]:

$$\frac{d\overline{\rho}}{dt} + \frac{\partial}{\partial x_j}\left(\overline{\rho}\widetilde{u_j}\right) = 0 \tag{2.3.5}$$

$$\frac{d}{dt}\left(\overline{\rho}\widetilde{u_i}\right) + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_i}\widetilde{u_j} + \overline{p}\delta_{ij} + \widetilde{\tau_{ij}^{tot}}\right] = 0 \tag{2.3.6}$$

$$\frac{d}{dt}\left(\overline{\rho}\widetilde{e_0}\right) + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j}\widetilde{h_0} + \widetilde{q_j^{tot}} - \widetilde{u_i}\widetilde{\tau_{ij}^{tot}}\right] = 0 \tag{2.3.7}$$

Where the stress tensor and heat flux consist of one turbulent part and one laminar part:

$$\widetilde{\tau_{ij}^{tot}} = \widetilde{\tau_{ij}^{lam}} + \widetilde{\tau_{ij}^{turb}} \tag{2.3.8}$$

$$\widetilde{\tau_{ij}^{lam}} = \widetilde{\tau_{ij}} = \nu\left(\frac{\partial\widetilde{u_i}}{\partial x_j} + \frac{\partial\widetilde{u_j}}{\partial x_i} - \frac{2}{3}\frac{\partial\widetilde{u_k}}{\partial x_k}\delta_{ij}\right) \tag{2.3.9}$$

$$\widetilde{\tau_{ij}^{turb}} = -\overline{\rho u_i'' u_j''} = \nu_t\left(\frac{\partial\widetilde{u_i}}{\partial x_j} + \frac{\partial\widetilde{u_j}}{\partial x_i} - \frac{2}{3}\frac{\partial\widetilde{u_k}}{\partial x_k}\delta_{ij} - \frac{2}{3}\overline{\rho}k\delta_{ij}\right) \tag{2.3.10}$$

$$\widetilde{q_j^{tot}} = \widetilde{q_j^{lam}} + \widetilde{q_j^{turb}} \tag{2.3.11}$$

$$\widetilde{q_j^{lam}} = \widetilde{q_j} \approx -C_p\frac{\nu}{Pr}\frac{\partial\widetilde{T}}{\partial x_j} \tag{2.3.12}$$

$$\widetilde{q_j^{turb}} = C_p\overline{\rho u_j'' T} \approx -C_p\frac{\nu_t}{Pr_t}\frac{\partial\widetilde{T}}{\partial x_j} \tag{2.3.13}$$

$$\overline{p} = \overline{p}\left(\overline{\rho}, \widetilde{e}\right) \tag{2.3.14}$$

2.3.1 Two equation models

The open turbulent momentum equation 2.2.18 is subtracted from the instantaneous momentum equation 2.2.8 and which yields the transport equation for the instantaneous velocity, u_i''. To get the transport equation

for the Reynold stresses, $\widetilde{\overline{\rho} u_i'' u_j''}$, the transport equation of the instantaneous velocity is multiplied with the instantaneous velocity, u_j'' and then Favre averaged. To get the transport equation for turbulent kinetic energy, k, the trace, $(i = j)$, of the transport equation for the Reynold stresses are taken and then divided by two $\overline{\rho}$ according to the definition for turbulent kinetic energy 2.2.21. For simplicity, all the terms related to pressure (pressure diffusion, pressure work and pressure dilatation) has been omitted since the Mach number in this thesis is well below the hyper sonic range.

$$\frac{\partial \overline{\rho} k}{\partial t} + \frac{\partial}{\partial x_j}\left[\overline{\rho}\widetilde{u_j} k - \overline{\tau_{jk} u_i''} + \tau_{ij}^{turb}\right] = -\overline{\rho u_j'' u_i''}\frac{\partial \widetilde{u_k}}{\partial x_k} - \overline{\tau_{ij}\frac{\partial u_i''}{\partial x_j}} \tag{2.3.15}$$

Where the two terms on the right hand side of equation 2.3.15, is production and dissipation of turbulent kinetic energy. For simplicity the Favre ($\widetilde{\cdot}$) and Reynolds ($\overline{\cdot}$) averaging notation will not be written in the following equations. The turbulence model used in this work has not been specially developed for compressible real gas flow. The equations of state for the real gas is given by a real gas property file, see 3.1 for more information.

2.3.2 Shear Stress Transport Model

The SST (Shear Stress Transport) model is an eddy-viscosity model. It utilises the original $k-\omega$ model of Wilcox [28] in the inner region of the boundary layer and for the outer region and free stream it uses the $k-\epsilon$ model. According to Menter, [18] the SST model avoids the strong free stream sensitivity that the Wilcox model has. The $k-\omega$ model is simple and therefore stable. It is also good in predicting the adverse pressure gradient flow which is one of the main weaknesses of the $k-\epsilon$ model. The $k-\epsilon$ model over predicts the adverse pressure gradient region because the dissipation is under predicted. It also needs near-wall modification, which introduces simplifications. Therefore the $k-\omega$ model is used in the viscous region, buffer region and log-law region which can be seen in figure 2.3.1.

But the disadvantage of the $k-\omega$ model is that it depends on the free stream value of ω [17]. When combining the two equations it is convenient to re-write the $k-\varepsilon$ equation into an $k-\omega$ equation by the relation $\omega \equiv \varepsilon/(\beta^* k)$. For simplicity the entire derivation will not be presented but can bee seen in [7]. The final equations for k and ω [18]:

$$\frac{D\rho k}{Dt} = \tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta^* \rho\omega k + \frac{\partial}{\partial x_j}\left[\left(\mu + \sigma_k \mu_t\right)\frac{\partial k}{\partial x_j}\right] \tag{2.3.16}$$

$$\frac{D\rho\omega}{Dt} = \frac{\gamma}{\nu_t}\tau_{ij}\frac{\partial u_i}{\partial x_j} - \beta\rho\omega^2 + \frac{\partial}{\partial x_j}\left[\left(\mu + \sigma_\omega \mu_t\right)\frac{\partial \omega}{\partial x_j}\right] + 2\rho\left(1 - F_1\right)\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} \tag{2.3.17}$$

The material derivative $\frac{D\Phi}{Dt}$, in 2.3.16 $\Phi \equiv (\rho k)$ and 2.3.17 $\Phi \equiv (\rho\omega)$, is defined:

$$\frac{D\Phi}{Dt} = \frac{\partial \Phi}{\partial t} + u_{ij}\frac{\partial \Phi}{\partial x_i} \tag{2.3.18}$$

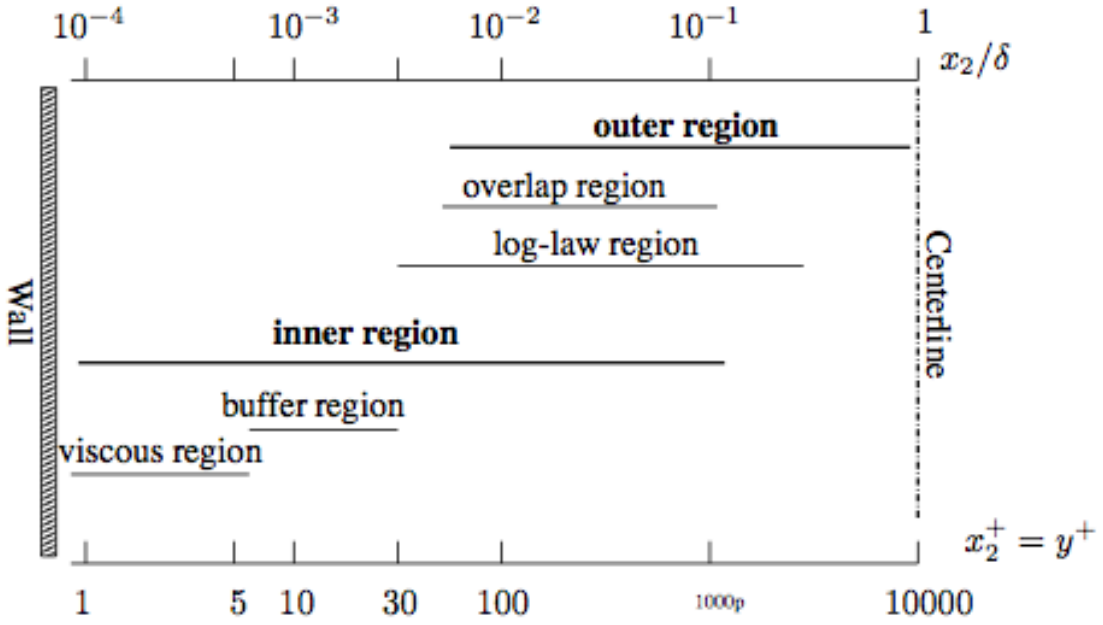


Figure 2.3.1 – *The near-wall region for $Re_\tau = 10000$. Half of the channel width is denoted by δ. x_2^+ denotes the normalized wall distance* Picture reprinted from [7]

The blending functions, F_1 and F_2, makes the SST model smoothly shift from $k-\omega$ values in the inner region of the boundary layer to $k-\varepsilon$ values in the outer region by going from 1 to 0. F_1 and F_2 is given by:

$$F_1 = tanh\left(arg_1^4\right), \qquad F_2 = tanh\left(arg_2^2\right) \tag{2.3.19}$$

$$arg_1 = min\left(\frac{500\nu}{y^2\omega}, \frac{4\rho k}{CD_{k\omega}y^2}\right), \qquad arg_2 = max\left(\frac{2\sqrt{k}}{\beta^*\omega y}, \frac{500\nu}{y^2\omega}\right) \tag{2.3.20}$$

The formulation for arg_1 in equation 2.3.20 ensures that F_1 is zero for $y^+ > 70$.

$$CD_{k\omega} = max\left(2\rho\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right) \tag{2.3.21}$$

Eddy viscosity is defines as:

$$\nu_t = \frac{a_1 k}{max\left(a_1\omega, \Omega F_2\right)} \tag{2.3.22}$$

The relation between constants for the original $k-\omega$ model, original $k-\varepsilon$ model, by [28], and the $SST\,k-\omega$ model, by [18], can be seen by the following relation:

$$\Phi = F_1\Phi_1 + (1-F_1)\,\Phi_2 \tag{2.3.23}$$

The constants are:

Table 2.3.1 – The original SST model constants for the set 1 (ϕ_1)

$\sigma_{k1} = 0.5$	$\sigma_{\omega 1} = 0.5$	$\beta_1 = 0.0750$	$a_1 = 0.31$
$\beta^* = 0.09$	$\kappa = 0.41$	$\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}}$	

Table 2.3.2 – The original SST model constants for the set 2 (ϕ_2)

$\sigma_{k2} = 1.0$	$\sigma_{\omega 2} = 0.856$	$\beta_2 = 0.0828$
$\beta^* = 0.09$	$\kappa = 0.41$	$\gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^*}}$

Implementation in CFX

The following equations and formulations are from [3].

$$\frac{\partial \rho k}{\partial t} + \frac{\partial}{\partial x_j}\left(\rho u_j k\right) = \frac{\partial}{\partial x_j}\left[\left(\mu + \mu_t\sigma_k\right)\frac{\partial k}{\partial x_j}\right] + P_k - \beta^*\rho k\omega + P_{kb} \tag{2.3.24}$$

$$\frac{\partial \rho \omega}{\partial t} + \frac{\partial}{\partial x_j}\left(\partial \rho\omega\right) = \frac{\partial}{x_j}\left[\left(\mu + \mu_t\sigma_\omega\right)\frac{\partial \omega}{\partial x_j}\right] + \left(1-F_1\right)2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j} + \alpha_3\frac{\omega}{k}P_k - \beta\rho\omega^2 + P_{\omega b} \tag{2.3.25}$$

$$P_{\omega b} = \frac{\omega}{k}\left(\left(\alpha+1\right)C_3 max\left(P_{kb}\right)\cdot sin\left(\phi\right) - P_{kb}\right) \tag{2.3.26}$$

The constants have been combined in a linear manner as equation 2.3.23 and are stated in table 2.3.1 and table 2.3.2.

$$arg_1 = min\left(max\left(\frac{\sqrt{k}}{\beta' y}, \frac{500\nu}{y^2\omega}\right), \frac{4\rho k}{CD_{k\omega}\sigma\omega 2y^2}\right) \tag{2.3.27}$$

Clearly seen in the formulation for arg_1, equation 2.3.27, has *CFX* further developed the model to ensure that F_1 goes to zero when $y^+ > 70$.

$$CD_{k\omega} = max\left(2\rho\frac{1}{\sigma_{\omega 2}\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 1.0\cdot 10^{-10}\right) \tag{2.3.28}$$

$$P_k = \mu_t\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\frac{\partial u_i}{\partial x_j} - \frac{2}{3}\frac{\partial u_k}{\partial x_k}\left(3\mu_t\frac{\partial u_k}{\partial x_k} + \rho k\right) \tag{2.3.29}$$

2.3.3 Near wall treatment

The normalized wall distance, also called y^+, is defined by [7]:

$$y^+ \equiv \frac{u_\tau x_2}{\nu} \tag{2.3.30}$$

The wall shear stress, τ_w, is the velocity gradient at the wall times the dynamic viscosity, [7]:

$$\tau_w = \mu\left.\frac{\partial \bar{v}_1}{\partial x_2}\right|_w \tag{2.3.31}$$

From the wall shear stress, τ_w, the wall friction velocity, u_τ, is defined [7]:

$$u_\tau = \left(\frac{\tau_w}{\rho}\right)^{1/2} \tag{2.3.32}$$

There are two possible ways of treating the boundary layer in turbulence modeling. One is resolving the near wall region, seen in figure 2.3.1, with a very fine computational mesh. The second way is to use wall functions. As can be seen in figure 2.3.1 the viscous sub layer is located right next to the wall and is very thin, $y^+ \leq 5$, it follows a linear relationship with the velocity. Then there is a buffer region, also seen in figure 2.3.1, located at right before the log-law region, $30 \leq y^+ \leq 100$, the upper limit is dependent of the Reynolds number. When resolving the near wall region the node adjacent to the wall should be below $y^+ = 1$ and preferably 30 nodes below y^+ of 100. According to [3] the y^+ value for heat transfer predictions need to be quite low preferably $y^+ = 1$. But in [18] it is stated that the $k-\omega$ model can be run with the first grid point as far out as $y^+ = 3$ without a deterioration of the results.

In this work the automatic wall functions in CFX were used. Since the SST model already gives an analytical expression for ω in the viscous sublayer, the main idea with the wall function formulation in CFX is to blend the wall value of ω between the logarithmic region and the near wall formulations.When automatic wall functions is used it is recommended to resolve the boundary layer with no less than 10 nodes, according to [2].

2.3.4 Convergence criteria

When solving the equations the solutions will have some error due to the iterative solution technique and the nature of the equations. This error is plotted as a residual in CFX. The residual give the difference between the left-hand side of the equation and the right-hand side. To make the scale of the residual useful the residual is divided by appropriate scale at the specific point. The RMS residual is taking this normalized residuals for the entire domain, squaring, taking the mean and then taking the square root of the mean. The solution is considered converged when the RMS (Root Mean Square) residuals have decreased below a value of $1 \cdot 10^{-4}$ according to [2] and stabilised around a value.

For this case the balance of the flow is also an convergence criteria. The residuals can go to a constant value but the total flow in and out of the domain is not taken into consideration. By studying the sum of flows for a given equation, mainly energy equation, on a particulate domain the simple law of what goes in must come out is satisfied. The imbalance need to be within a value of $\pm 0.5\%$ to ensure that the convergence is satisfied.

2.4 Heat transfer

To be able to solve the heat transfer the energy equation 2.2.9 is solved together with the continuity equation 2.2.7 and the momentum equation 2.2.8. In order to obtain any heat transfer between two medias a temperature difference need to be present. There are three different types of heat transfers, conduction, convection and radiation. Both conduction and convection is treated in this thesis but the radiation is neglected.

Convection

Occurs when a solid surface is in contact with a fluid, either the surface is hotter than the fluid and results in a temperature increase in the fluid or the fluid is hotter than the surface and gives a temperature increase in the surface. This is called natural convection. Forced convection occurs when apparatus such as a fan or pump drives the convection.
The convective heat transfer from a fluid to a solid can be described by heat flux [11]:

$$\dot{Q}_c = h_g(T_{aw} - T_{wh}) \tag{2.4.1}$$

The *adiabatic wall temperature*, T_{aw}, is defined by taking the temperature at the wall in a moving fluid stream when there is no heat transfer between the wall and the stream. One dimensional heat transfer through a solid wall can be described by [11]:

$$\dot{Q}_w = \frac{k_w}{\Delta L}(T_{wh} - T_{wc}) \tag{2.4.2}$$

The heat transfer through the coolant liquid is described by [11]:

$$\dot{Q}_L = h_L(T_{wc} - T_L) \tag{2.4.3}$$

For steady state heat flow from hot gas can be written [11]:

$$\dot{Q} = \dot{Q}_c + \dot{Q}_r = \dot{Q}_w = \dot{Q}_L \tag{2.4.4}$$

Radiation heat flux $\dot{Q}_r$ is neglected and the equation simplifies to

$$\dot{Q} = \dot{Q}_c = \dot{Q}_w = \dot{Q}_L \tag{2.4.5}$$

Conduction

The transfer of energy from more energetic particle to a less energetic particle yields a higher temperature.

2.5 Real gas

For this thesis the fluid used is hydrogen. It is known that each hydrogen molecule (H_2) consists of two atoms. Molecular hydrogen can occur in two isomeric forms, one with the protons spinning aligned parallel (orthohydrogen) and the other with the protons spinning aligned antiparallel (parahydrogen). The parahydrogen is the most common natural occurring form and the form of hydrogen used for fuel in some space rockets. When implementing the material, parahydrogen, in CFX a real gas property file (RGP) is used and this file is created by a software called REFPROP (REFerence fluid PROPerties). The program use three models for the thermodynamic properties of pure fluids. The equation of state explicit in the Helmholtz energy, the modified Benedict-Webb-Rubin equation of state and the ECS (Extended Corresponding States) model [15]. For more information about the software please see [15].

2.6 Subdomains

The subdomain is a 3D region used to specify values for volumetric sources in CFX. The subdomain allow sources of energy, mass, momentum, radiation, components and turbulence to be specified. The subdomain can only be created inside a 3D domain. When modeling porous momentum losses, a momentum loss due to an obstruction in the flow direction, such as honeycombs, porous filters and so on, the isotropic and directional loss models are useful. The CFX Modeling guide [2] suggests that the subdomains can be used when modeling

effects of flow straightening devices such as honeycombs, porous plates and turning vanes. In this case the subdomain is used to model flow resistance.

For this specific case the isotropic loss model is used since it is this model the in-house code TCCOOL uses for modeling the resistance of the flow. When using the isotropic loss model it simply adds a source term to the momentum equation, 2.2.8:

$$S_{Mi} = -k_{loss}\frac{1}{2}\rho u_i^2 \tag{2.6.1}$$

So the momentum equation 2.2.8 will look like:

$$\frac{d}{dt}\left(\rho u_i\right) + \frac{\partial}{\partial x_j}\left(\rho u_i u_j + p\delta_{ij} - \tau_{ij}\right) + S_{Mi} = 0 \tag{2.6.2}$$

3 Method

The work was gradually increased in difficulty. Starting with a simple straight channel and progressing to a system with a inlet manifold, 5 tubes and outlet manifold. The workflow is described from the chart in figure 3.0.1. The work started in the top of the chart by testing the subdomain. Parallel to testing different meshes they where also validated against the *TCCOOL* software. Both the subdomain and the porous domain where tested with different mesh resolutions in order to see if any of the two domains could take a lower mesh resolution than the other but still produce accurate pressure loss and temperature increase. When the mesh for the validation case where set the work continued to the interface investigation where both from fine to coarse mesh interfaces where tested to coarse to fine interfaces. Then the inlet manifold were placed on one tube followed by the inlet manifold placed on five tubes in a row. Next a outlet manifold where placed and it was suggested that a varying k_{loss} should had been set in order to produce a more accurate pressure loss but this was in fact only tested on the validation case. The heat flux where set to vary depending on the mass flow after the simulation with five tubes inlet - and outlet manifold worked. All of these cases will be presented, in the order that they were preformed, in this chapter.

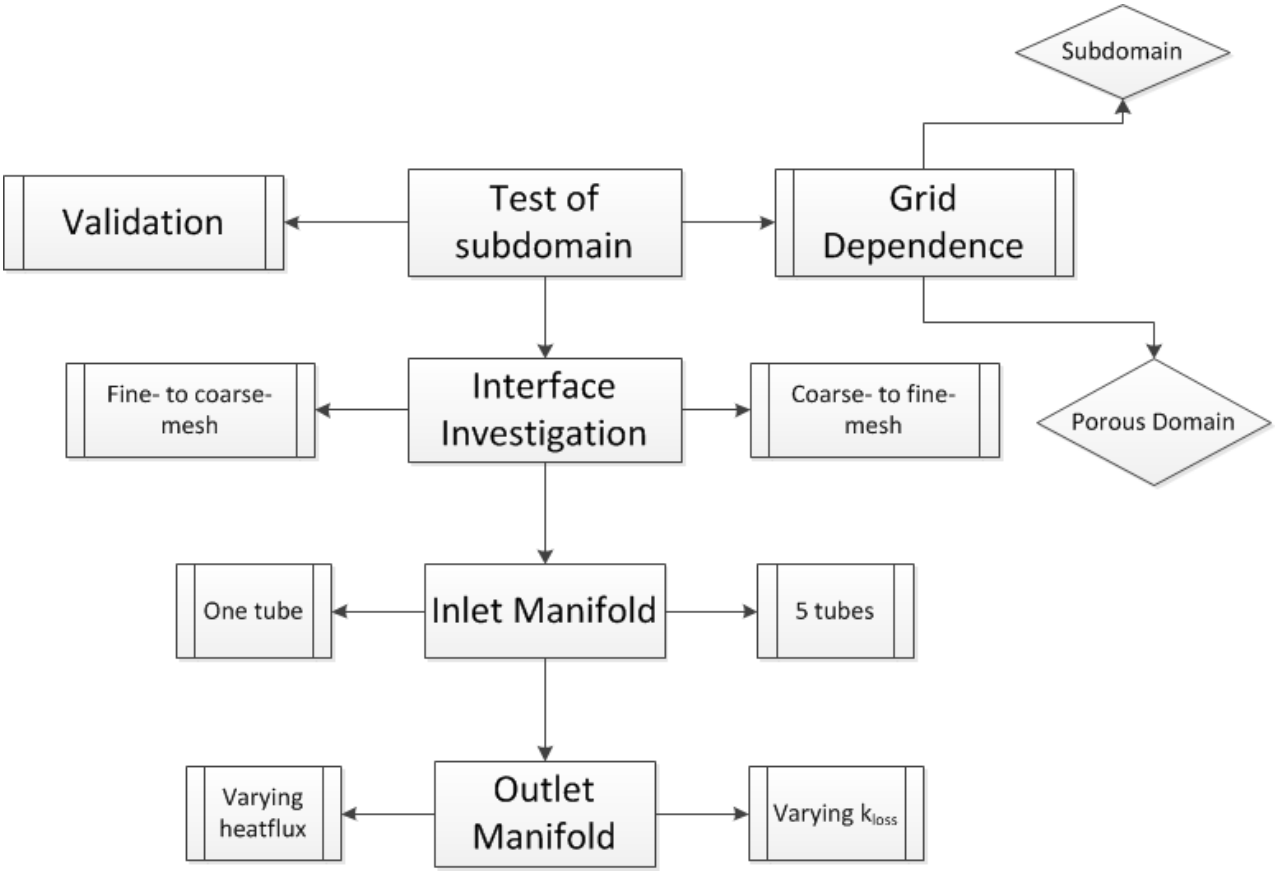


Figure 3.0.1 – *The flowchart for the work of the different parts of the simulation process. How the different parts came together into the final results, system of inlet manifold 5 tubes and outlet manifold with varying heatflux and* k_{loss}.

Softwares

The full name of the softwares used are stated in appendix A.

General Boundary Conditions

All walls in the model are adiabatic, meaning no heat is transferred out from the walls or into the walls, except for the wall where the heat flux is applied, the following chapters will describe what wall this is. When stating wall it is merely a surface where the specific boundary condition is applied on. No solid walls are present. When a coarse mesh is present a subdomain is applied in that mesh and in order to model the friction loss the walls are set to be free slip so no influence of the walls are present.

Numerical Method

The simulations are stationary and steady state. The turbulence model used were $SST\ k-\omega$ model by [18] described in chapter 2.3.2 together with a wall function, described in chapter 2.3.3. The Total Energy equation 2.2.3 together with the viscous work term, which is the last term in eq. 2.2.3, have been selected to model the heat transfer. Second order upwind scheme was used to discretize the equations, for the turbulence numerics the first order upwind scheme were utilized. *CFX* is a fully coupled, implicit solver so no pressure-velocity coupling is needed as in solvers like *Fluent*.

3.1 Material Properties

In the simulation the fluid is only modeled. No solid material is present in the simulations, such as solid walls. The fluid used for this thesis is parahydrogen and for the equation of state the software REFPROP has been used to produce an Real Gas Property file, later called rgp-file. Evaluating fluid property through a complex equation of state during a solution can be very computationally demanding. Therefore the software REFPROP has been used to produce an rgp-file for the parahydrogen. The NIST database via the program REFPROP described in chapter 2.5 supports the dry superheated vapor model. The superheated region is represented by nine tables as a function of temperature and pressure, the range of the temperature variations and pressure variations is set by the user of the software. For this thesis the temperature range where $26[K] < T < 1200[K]$ with increments of $1[K]$ and the pressure range where $10000[Pa] < P < 5 \cdot 10^7[Pa]$ with increments of $5 \cdot 10^4[Pa]$ which gives a matrix of 1174x999 for each variable. The following variables that is listed in the rgp-file, each number representing the table number, are :

1. $h(P,T)$ Specific enthalpy
2. $c(P,T)$ Speed of sound
3. $\nu(P,T)$ Specific volume
4. $c_\nu(P,T)$ Specific heat at constant volume
5. $c_p(P,T)$ Specific heat at constant pressure
6. $\frac{\partial P}{\partial \nu}_T(P,T)$ Partial derivative of pressure with respect to specific volume at constant temperature
7. $s(P,T)$ Specific entropy
8. $\mu(P,T)$ Dynamic viscosity

9. $\lambda(P,T)$ Thermal conductivity

(3.1.1)

3.2 Validation

Since the purpose of the thesis is to investigate if it is possible to use a very coarse mesh for the flow in the cooling channels and only resolve the inlet- and outlet manifold the case presented is tested for a very coarse mesh. The mesh used in this validation case, figure 3.2.1, are not supposed to resolve the boundary layer and is only supposed to take up the heat load and produce a pressure drop. Therefore the only parameters needed to be validated is the pressure into the domain and temperature out of the domain. The case has a domain which is $4.2[m]$ long and has a quadratic cross section of $3.2 \cdot 3.2 = 10.24[mm^2]$, specifics about the mesh can be seen in table 3.2.1.

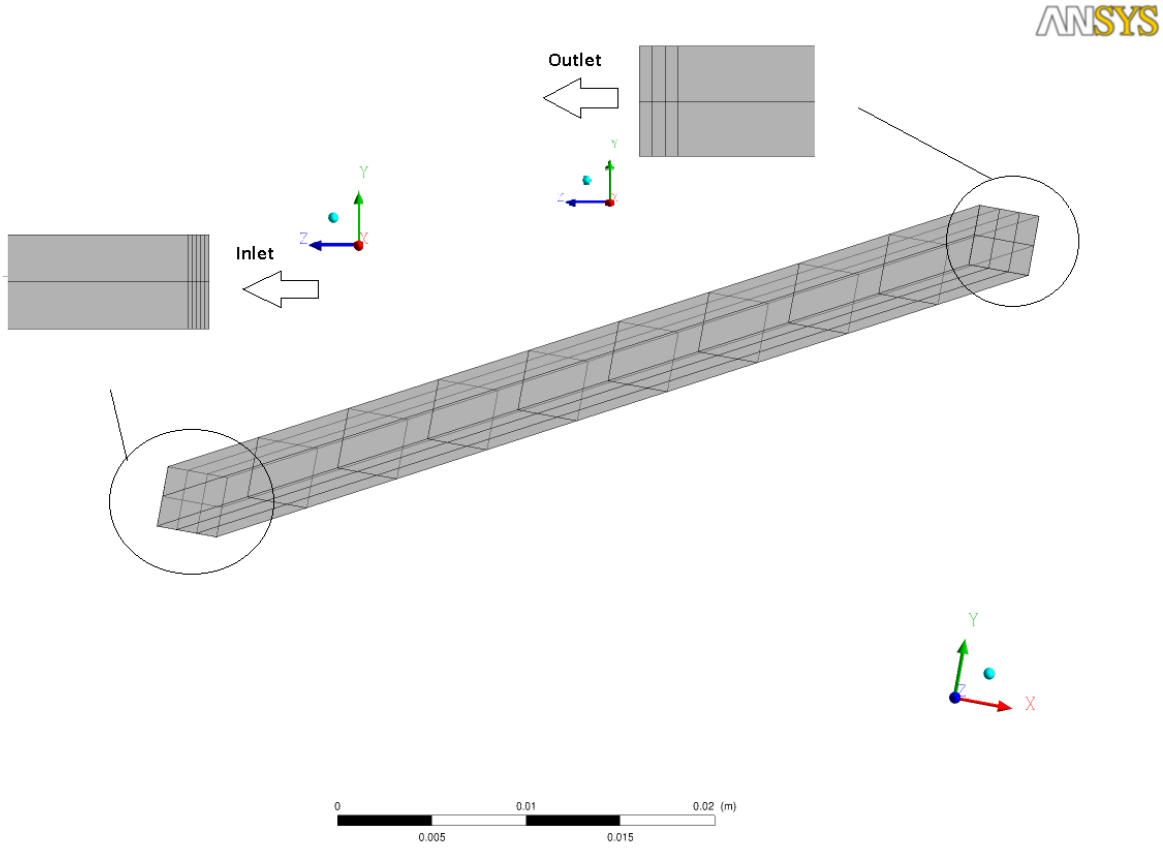


Figure 3.2.1 – *Mesh for validation simulation*

Table 3.2.1 – Mesh statistics for the validation case

Number of nodes:	216
Number of elements:	102
	Hexahedra elements
Total Volume:	$4.3008 \cdot 10^{-5}[m^3]$

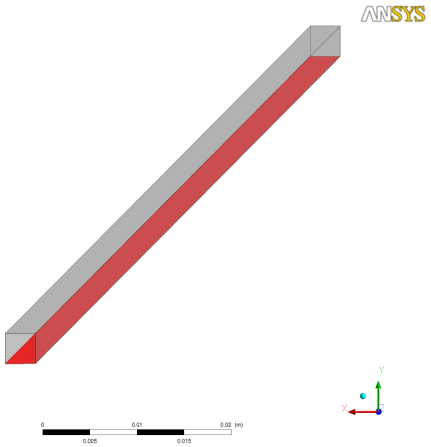


Figure 3.2.2 – *Red surface shows "flamewall" where the heat flux is applied for validation simulation, note here the inlet is in the foreground.*

The heat flux is applied on the flamewall which is highlighted in figure 3.2.2. The main parameters that were of interest of in the validation were the temperature and pressure loss over the domain. In fact, only the temperature out and the pressure on the inlet need to correlate with the TCCOOL simulation in order to consider the case validated, since the boundary condition for inlet temperature and outlet pressure is set.

3.2.1 Software: TCCOOL

The in-house code *TCCOOL* were used to validate the simulation of the case described in this chapter. The heat load from the flame is supplied by the TDK (Two Dimensional Kinetics) code, for more information about this software please see `http://www.seainc.com/`. The *TCCOOL* software is similar to the software *RTE2002* (Rocket Thermal Evaluation) that was developed by Tara Technologies via funding by NASA Lewis research center [19].
The *RTE2002* software conjugates all thermal/fluid processes in the propulsion system of a rocket engine in order to obtain the matched results for the thermal fields. The processes include convection and radiation heat transfer from hot combustion gases to the liner of the engine; convection to the coolant and conduction heat transfer with walls.
The inputs to *RTE2002* is the composition of fuel/oxidant mixtures and flow rates; chamber pressure; coolant entrance temperature and pressure; dimensions of the engine; materials and number of nodes in different parts of the engine. It handles temperature variations in axial, radial and circumferential directions. The results are listing of nodal temperatures, rates of heat transfer and hot-gas and coolant thermal and transport properties. The main difference between *RTE2002* and *TCCOOL* is that *TCCOOL* has other in house correlations for heat transfer coefficients. For more information about the numerical methods please see [19].

Boundary conditions

Volvo Aero has given the initial boundary conditions for the simulation in TCCOOL written in table 3.2.2. All the following boundary conditions are therefore based on these values. The mass flow specified is for 456 tubes. In the following *CFX* simulations the mass flow is divided by 456 because only one channel is simulated.

Table 3.2.2 – Boundary conditions for TCCOOL simulation.

Mass flow at inlet:	2.39 [kg/s]
Coolant Fluid temperature at inlet:	40 [K]
Coolant pressure at inlet:	57.1 [bar]

3.2.2 Domain

When choosing the domain to be used to implement the coefficient of loss *CFX* offers two choices. One is the subdomain, explained in chapter 2.6, and the other choice is changing the domain type from a fluid domain to a porous domain. The porous domain choice works similar to the subdomain in the theory in this work.
The porous domain choice was tested with a lower grid resolution, since the interest is to find a model that work with as low mesh resolution as possible. But it was seen that the porous domain could not handle a lower resolution than the subdomain and gave the same result as the subdomain, see chapter 4.1.1. Therefore the subdomain were chosen for all following simulations due to its flexibility, the possibility to only place the subdomain over a part of the domain instead of, as in the porous domain choice, placing it over the full fluid domain.

3.2.3 Mesh dependence

Seven different mesh resolutions were tested to see how low mesh resolution were possible to have but still receive correct temperature increase and pressure loss. The meshes can be seen in appendix C.

Boundary conditions

Table 3.2.3 – Boundary conditions for the mesh dependence investigation

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241229[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$1[atm]$
Outlet	Static pressure	$26.7[bar]$
Walls	Adiabatic	
	No slip	
Flamewall	Heatflux	Interpolation data according to figure 3.2.4
	No Slip	

3.2.4 Boundary Conditions

The table below states the boundary conditions for two simulations. Explained later in the chapter under **Heatflux** the two different heatfluxes are because in the first case, the heatflux were integrated from figure 3.2.4 by the trapetzoidal rule, introducing some numerical error, but in the second case the heatflux were simply taken from the *TCCOOL* interpolated result.

Heatflux

When applying the boundary condition for the flamewall, *CFX* can only place values on nodes, the value given closest to a node is interpolated. This means that even if a very precise list of heat flux values are given *CFX* only takes the values and then interpolates it to a node. Therefore it is not possible to use the very precise list of values given from the *TCCOOL* simulation, see figure 3.2.4, because the variations in heatflux is smeared in *CFX* as can be seen in figure 3.2.3 when comparing it to figure 3.2.4. Therefore the total heat flux is applied over the flame wall as terms of $[W/m^2]$. The smearing is because *CFX* interpolate by taking the three values

Table 3.2.4 – Boundary conditions for the validation cases with one tube

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241229[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Free slip	
Flamewall	Heatflux	$q_j = 5.1495[MW/m^2]$
		$q_j = 5.174409[MW/m^2]$
	Free Slip	

closest to the node and calculates a distance weighted value according to [2].

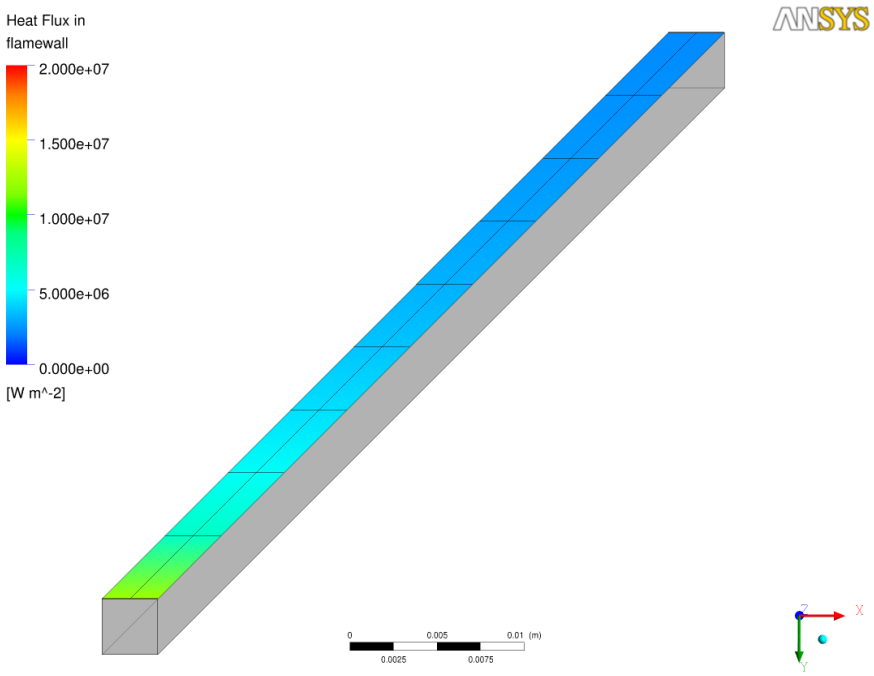


Figure 3.2.3 – *Contour plot from CFX when implementing the heat flux over the flame wall surface, this is the boundary condition applied on the wall and then visualised with a contour plot in CFX, the tube is seen from upsidedown.*

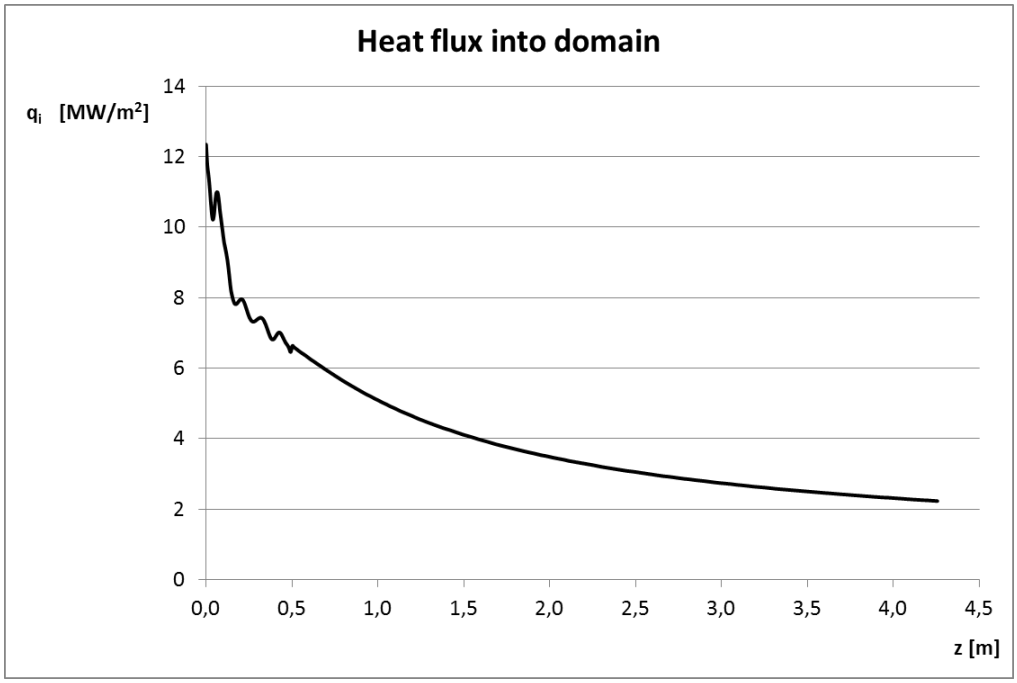


Figure 3.2.4 – *The accurate heat flux from the TCCOOL simulation, z-coordinate along the length of the tube.*

The total heat flux for one channel is calculated from the *.load*-file from the *TCCOOL* simulation column QW, see figure 3.2.4, with the trapezoidal rule.

$$Q_{one\,channel} = \int_0^L QW(L)dL = \sum_{a=0}^{L} \left((l(a+1) - l(a)) \frac{QW(a) + QW(a+1)}{2} \right) \tag{3.2.1}$$

Divided by the length of the wall $4.2[m]$ and scaled with $4.0[mm]/3.2[mm]$, where the scale is the difference between the width of the tube used in the *TCCOOL* simulation, when simulation includes the solid wall, and the simulations in this thesis where only the fluid is simulated, the constant heat flux is $q_j = 5.1495[MW/m^2]$. It is also possible to use the total heat flux value given in a main result file from the *TCCOOL* simulation this value was also tested in a simulation. From the *TCCOOL* simulation the total heat flux is $Q = 31.71209[MW]$ given in the *.par*-file. Divided by the number of channels, 456, and the area of the flamewall, $0.0032[m] \cdot 4.2[m] = 0.01344[m^2]$

$$q_j = \frac{Q}{n \cdot A_s} = \frac{31.71209}{456 \cdot 0.01344} = 5.174409[MW/m^2] \tag{3.2.2}$$

For simplicity the simulation with heat flux $q_j = 5.1495[MW/m^2]$ is called **Validation case 1** and the simulation with heat flux $q_j = 5.174409[MW/m^2]$ is called **Validation case 2**. For the following simulations where a constant heat flux value is used the first value of $q_j = 5.1495[MW/m^2]$ was chosen, see chapter 3.3.1, chapter 3.3.2 and chapter 3.4.1 and chapter 3.4.2. But when the heatflux became dependent of the mass flow it was simplest, and also more correct, to use the total heat flux from the *TCCOOL* simulation, see chapter 3.4.2 and chapter 3.5.

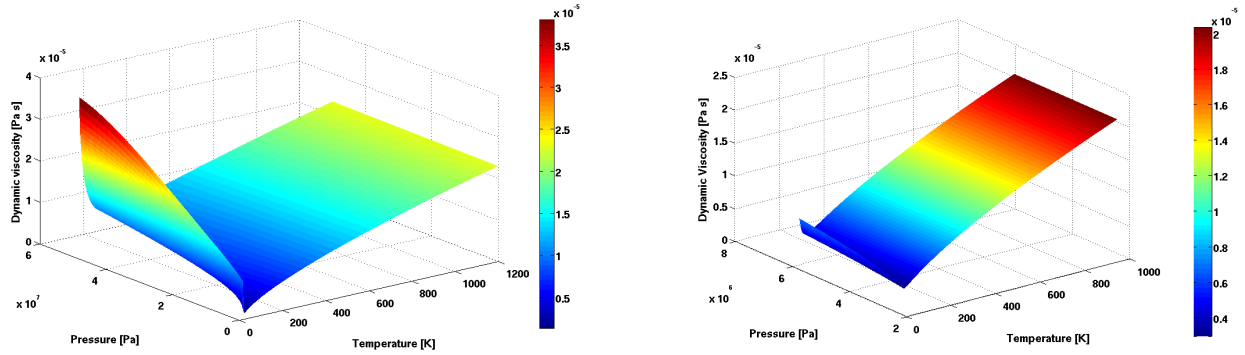
Coefficient of loss depending of flow, $k_{loss}(Re, \varepsilon_R)$

Similar as to the boundary condition, $q_j(\dot{m})$, the coefficient of loss k_{loss} vary depending of Reynolds number and roughness of the pipe, explained in chapter 2.1. The Reynolds number can be rewritten from the equation 2.1.10 with the mass flow to:

$$Re = \frac{4 \cdot \dot{m}}{\mu \cdot O} \tag{3.2.3}$$

Where O is the perimeter of the tube and μ is the dynamic viscosity. In CFX some expressions need to be created to be able to define a varying k_{loss}. In this case the k_{loss} will vary in each cell. But since the equation for k_{loss} is based on a one dimensional theory k_{loss} should only vary in the cross section of the channel normal to the streamwise direction. The difference should be expected to be small since the friction factor only depend on Reynolds number and the Reynolds number only vary with the dynamic viscosity, due to the .rgp-file. Seen in figure 3.2.5 the dynamic viscosity vary the most in the outer regions, where pressure and temperature are high and low. In the regions where this case is operating the dynamic viscosity vary between $2.9653 \cdot 10^{-6}[Pa \cdot s]$ and $2.0339 \cdot 10^{-5}[Pa \cdot s]$. So the error of the friction factor is assumed to be low.

Together with the expressions for the Serghides friction factor equations, see equations 2.1.6 to 2.1.9, and the definition for the Reynolds number, equation 3.2.3, an expression for k_{loss} can be defined. This varying k_{loss} were tested in the end of the project and therefore the simulations that are explained in the following chapters



(a) *Full range* **(b)** *Case range*

Figure 3.2.5 – *Surface plots for the dynamic viscosity used in the simulation, full range is taken from the .rgp-file, working range is the .rgp-file clipped to the current range.*

were not created with this varying k_{loss}. Only this validation cases were tested but it is expected to work the same as for all the following simulations.

3.3 Interface Investigation

Where it is desired to have fully resolved flow a fine mesh is used. This chapter explain the method for the investigation of what happens when having two very different meshes and combining them.
The investigation was conducted in two simulations, starting with one interface (fine-to-coarse), and progressing with two interfaces (fine-coarse-fine). When using the mesh from the validation case a jump, in some variables, between the fine mesh and the coarse mesh were observed. It was needed to refine the inlet in flow direction to ensure a smooth transition between the fine mesh and coarse mesh.

3.3.1 Fine to coarse simulation

This simulation required two different meshes since the first mesh was used to resolve the flow fully and the second, coarse, mesh were only supposed to give the correct temperature out and correct pressure drop over the domain, similar to the validation case.
The geometry used for the fine to coarse simulation can be seen in figure 3.3.1. Region 1 is $1[m]$ long and fully resolved and region 2 is $3.2[m]$ with a coarse mesh and a subdomain covering the full domain.

Fine mesh for "inlet and outlet"-region

The mesh used for the fully resolved region is shown in figure 3.3.2. Since this region will try to resolve the boundary layer, unlike the coarse region, the requirements are a bit higher. The mesh statistics are presented in table 3.3.1:

Table 3.3.1 – Mesh statistics for the fine mesh in the interface investigation case

Number of nodes:	102400
Number of elements:	95139
	Hexahedra elements
Total Volume:	$1.024 \cdot 10^{-5}[m^3]$

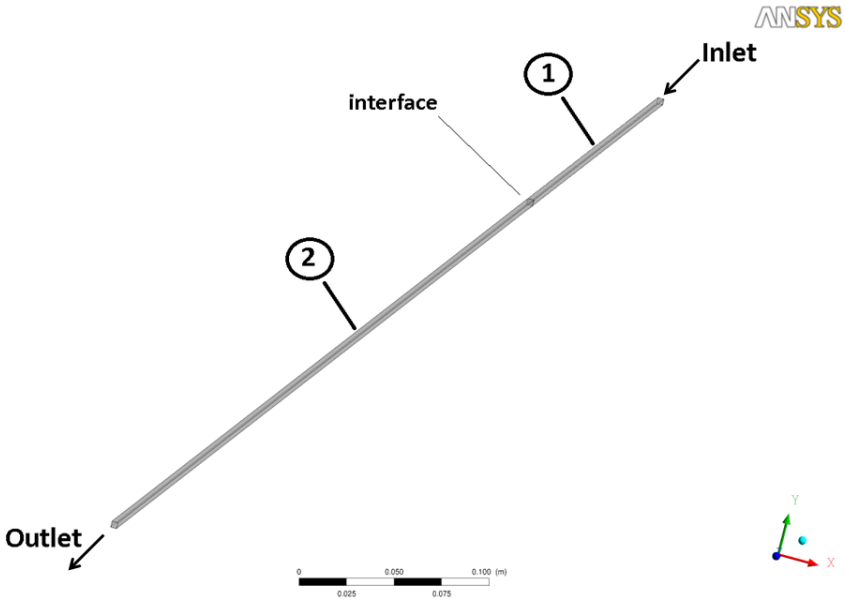


Figure 3.3.1 – *Domain for the simulation where the mesh goes from fine to coarser. Region 1 is fully resolved, region 2 is the where the coarse mesh is placed.*

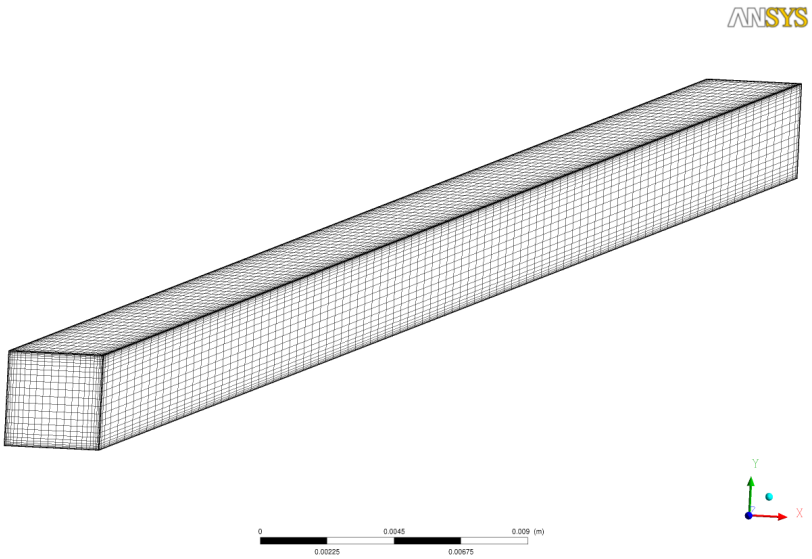


Figure 3.3.2 – *Mesh for region 1 seen in figure 3.3.1*

Coarse mesh for "middle"-region

As can be seen in figure 3.3.3 the mesh used for the coarse region is very coarse. The mesh goes from very fine to a very coarse resolution in flow direction the mesh needs to be gradually increased. A stretching is applied in the interface region of the coarse mesh, which can be seen in figure 3.3.4 and in the outlet region seen in figure 3.3.5. The mesh statistics are presented in the table 3.3.2:

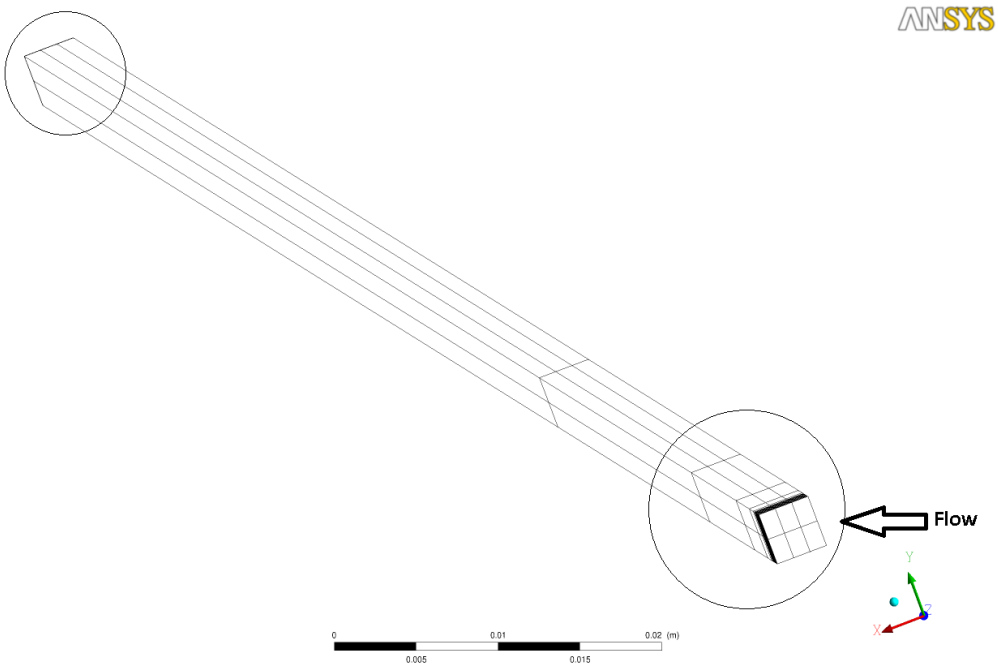


Figure 3.3.3 – *Mesh for region 2 seen in figure 3.3.1, the circle in the right corner represent the interface region zoomed up in figure 3.3.4 and the circle in the left corner represent the outlet region 3.3.5*

Table 3.3.2 – Mesh statistics for the coarse mesh

Number of nodes:	240
Number of elements:	114
	Hexahedra elements
Total Volume:	$3.2768 \cdot 10^{-5}[m^3]$

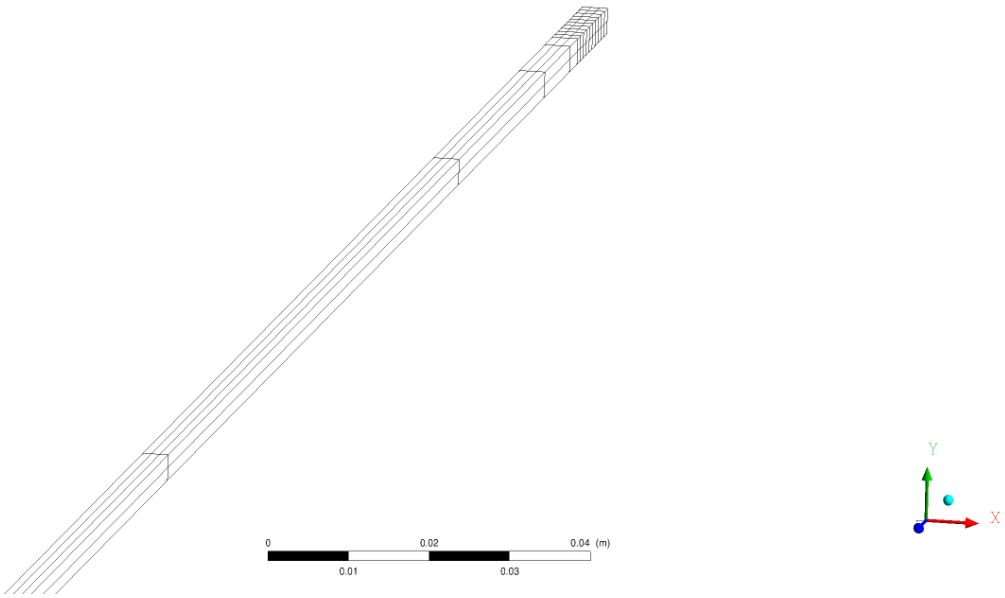


Figure 3.3.4 – *Close up of interface region seen in figure 3.3.3, from the circle in the lower right corner*

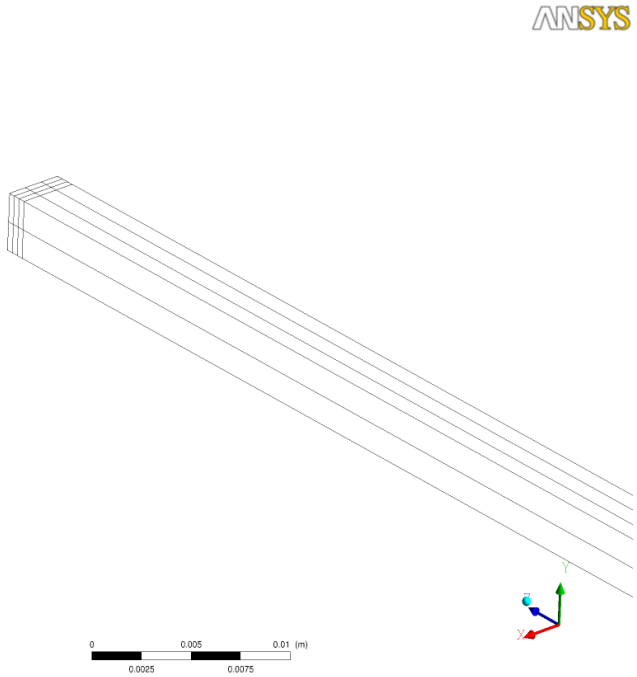


Figure 3.3.5 – *Close up of outlet region seen in figure 3.3.3, from the circle in the upper left corner*

Boundary Conditions

Table 3.3.3 – Boundary conditions for interface investigation of the fine to coarse interface simulation

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241228[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall
Flamewall	Heatflux	$q_j = 5.1495[MW/m^2]$
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall

3.3.2 Fine to coarse to fine simulation

Seen in figure 3.3.6 the domain consist of three meshes. The three mesh for this simulation is the same two meshes explained in chapter 3.3.1. The first mesh, label 1 in figure 3.3.6 is the same mesh as in figure 3.3.2. Mesh 3 in figure 3.3.6 is simply the same fine mesh used in region 1 but rotated so the interface side is matched to the interface side of the coarse mesh and the coarse mesh is the same as for the previous simulation chapter 3.3.1. The domain is in total 5.2$[m]$ long.

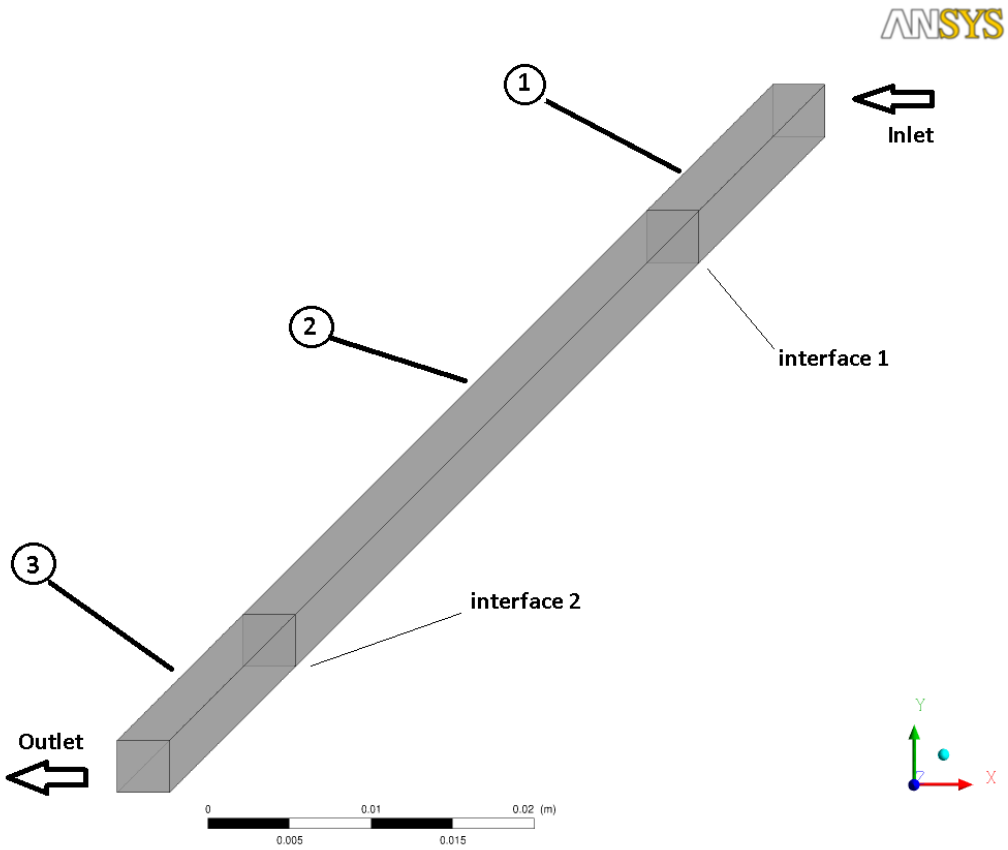


Figure 3.3.6 – *Domain for the simulation where the mesh goes from fine to coarser and then to a fine region. Region 1 and 3 is fully resolved while region 2 is modeled with a subdomain.*

Boundary Conditions

The simulation is very similar to the previous one. Also the boundary conditions are the same as for the previous simulation. The main difference is that a fine mesh is added, creating an interface where the flow goes from a coarse mesh to a fine mesh.

Table 3.3.4 – Boundary conditions for interface investigation of the fine to coarse to fine interface simulation

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241228[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall
Flamewall	Heatflux	$q_j = 5.1495[MW/m^2]$
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall

3.4 Inlet Manifold

In order to test the subdomain for a more complex flow situation closer to how the SCENE nozzle figure 1.0.1 cooling channel flow looks like, a concept geometry were created, with a system of five tubes together with an inlet manifold and a outlet manifold. In the first simulation only one tube with a inlet pipe was used see figure 3.4.1, when it was confirmed that the simulation converged the model were further developed with five tubes, see figure 3.4.4. The mass flow specified for the inlet are not the conditions as for the *TCCOOL* simulation specified by Volvo Aero, a lower mass flow is used. Although the mass flow is too low the results are still of interest since it were investigated if the solution could converge and how the subdomains work.

3.4.1 One tube simulation

The mesh is divided in two parts, one where the inlet region is resolved and the flow is turned into the tube from the manifold, explained later in this chapter. The second part of the mesh is a coarse mesh which is enabled by the subdomain. The full geometry can be seen in figure 3.4.1, the length of the tube is still 4.2$[m]$, as in the previous simulation, but now a inlet manifold is extruded from the geometry with a length of 1$[m]$. The cross section of this inlet manifold was calculated to be $9.0 \cdot 3.2[mm]$ in order to not get a Mach number higher than 0.3.

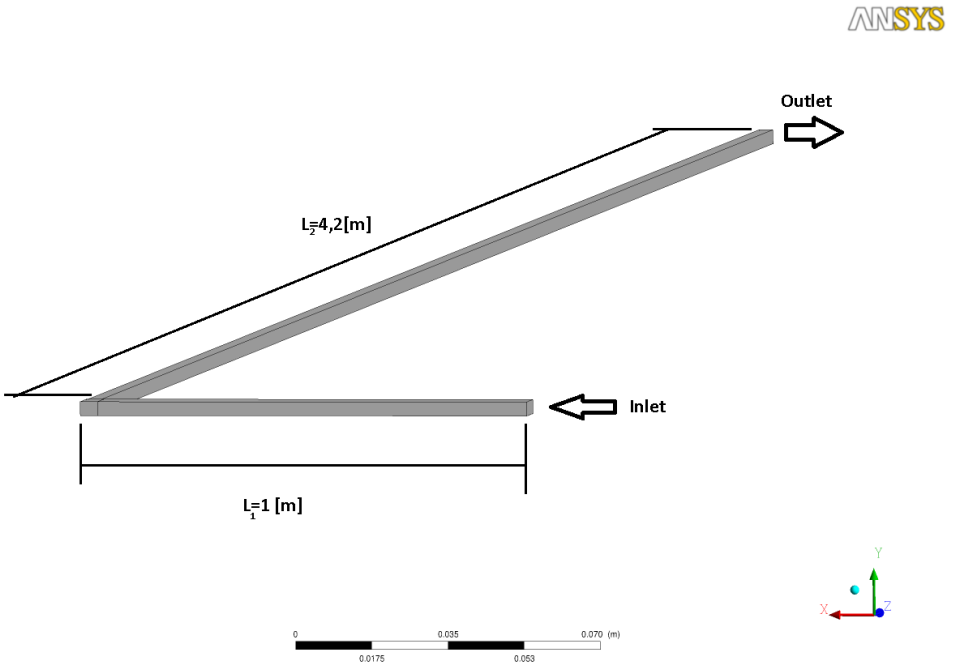


Figure 3.4.1 – *Geometry of the inlet manifold and one tube fine mesh*

Inlet region

The inlet region is fully resolved as in previous simulations in order to capture the more complex flow behavior.

Table 3.4.1 – Mesh statistics for the fine mesh of the inlet region, figure 3.4.2

Number of nodes:	178752
Number of elements:	166439
	Hexahedra elements
Total Volume:	$1.33894 \cdot 10^{-5}[m^3]$

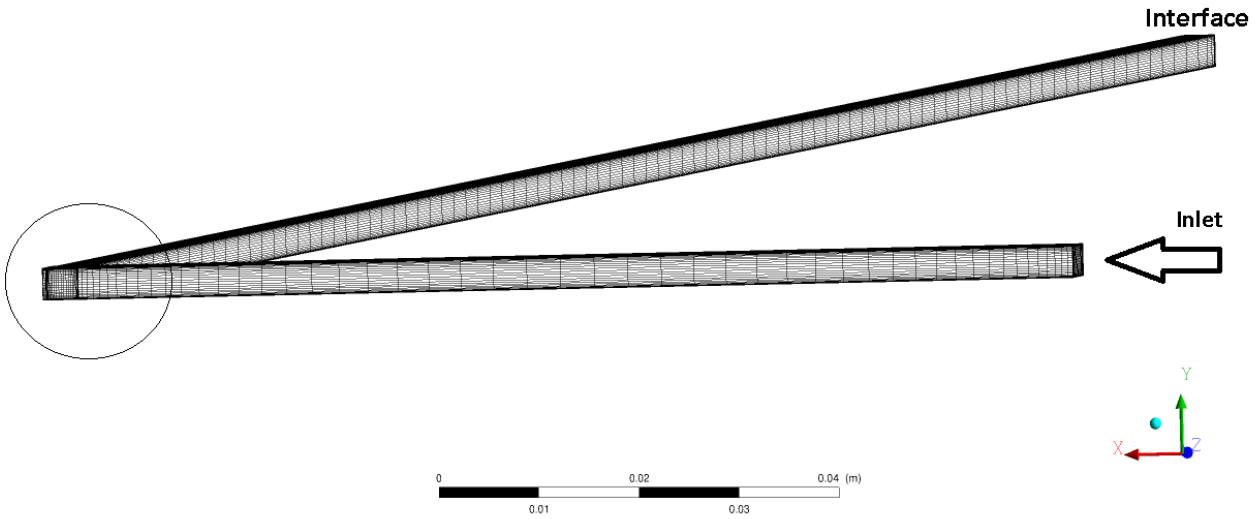


Figure 3.4.2 – *Mesh for the one tube with inlet manifold simulation, only fine region is shown.*

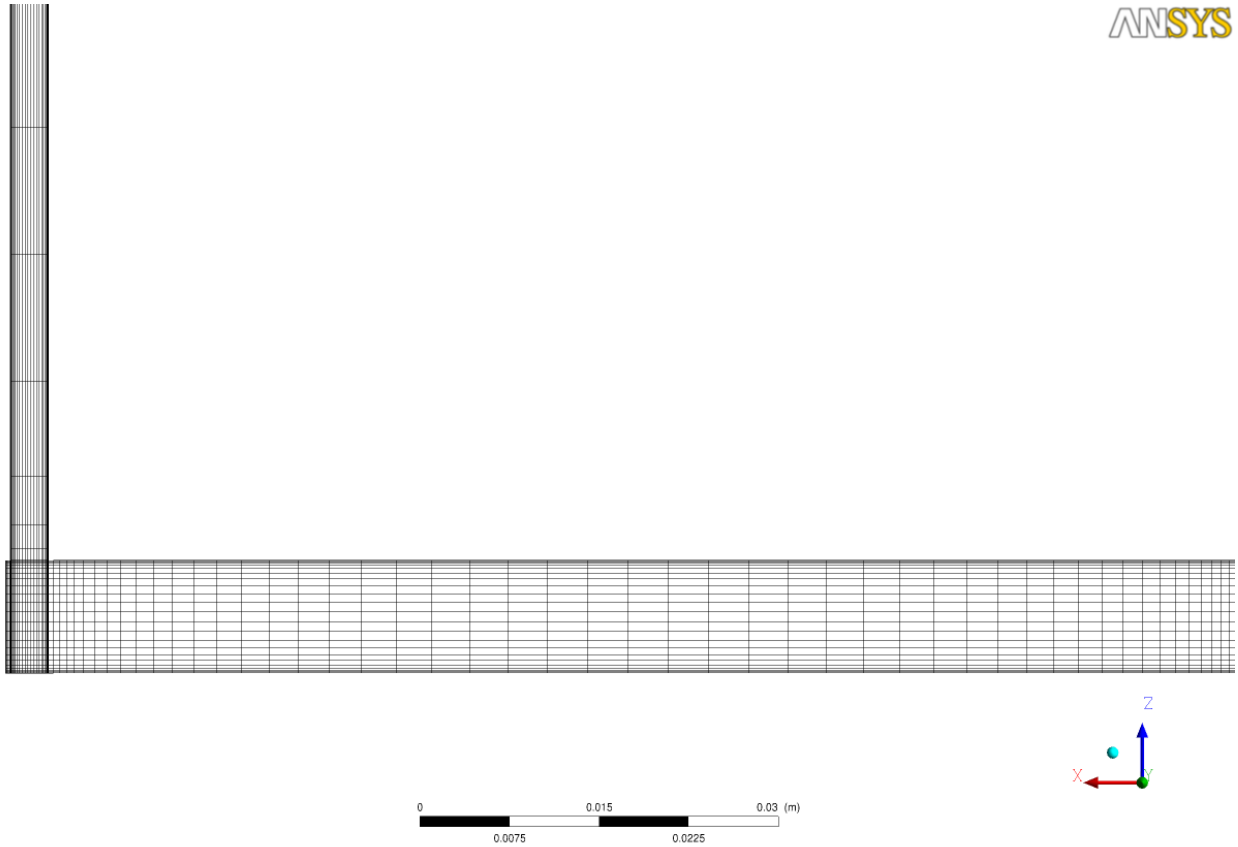


Figure 3.4.3 – *Close up of the region with a circle around it in figure 3.4.2, seen from above.*

Coarse region

The coarse mesh for this region is the same used in the previous simulation, see chapter 3.3.1.

Boundary Conditions

Table 3.4.2 – Boundary conditions for simulation with inlet manifold for one tube.

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241228[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall
Flamewall	Heatflux	$q_j = 5.1495[MW/m^2]$
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall

3.4.2 Five tube simulation

The mesh for this simulation is divided in several parts. The first tube (the farthest to the right in figure 3.4.4) is the same as in the previous simulation, figure 3.4.2, and the mesh used for the four other tubes are the same as the mesh seen in figure 3.4.2 but without the inlet manifold. The fine mesh for the four other tubes are seen in figure 3.4.5. Interfaces (the green arrows) between, see figure 3.4.4.

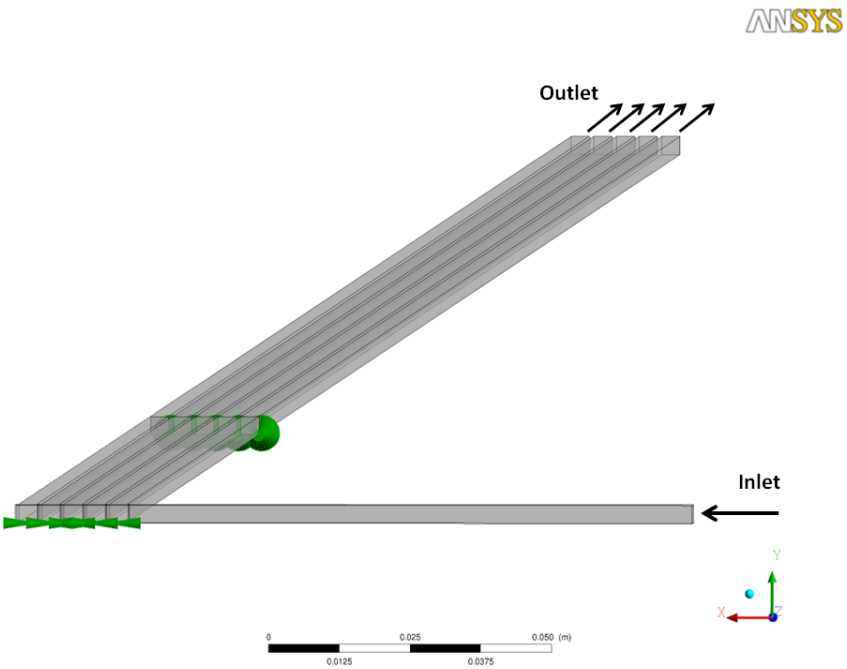


Figure 3.4.4 – *Domain for 5 tube with inlet manifold simulation, the green arrows show the interfaces*

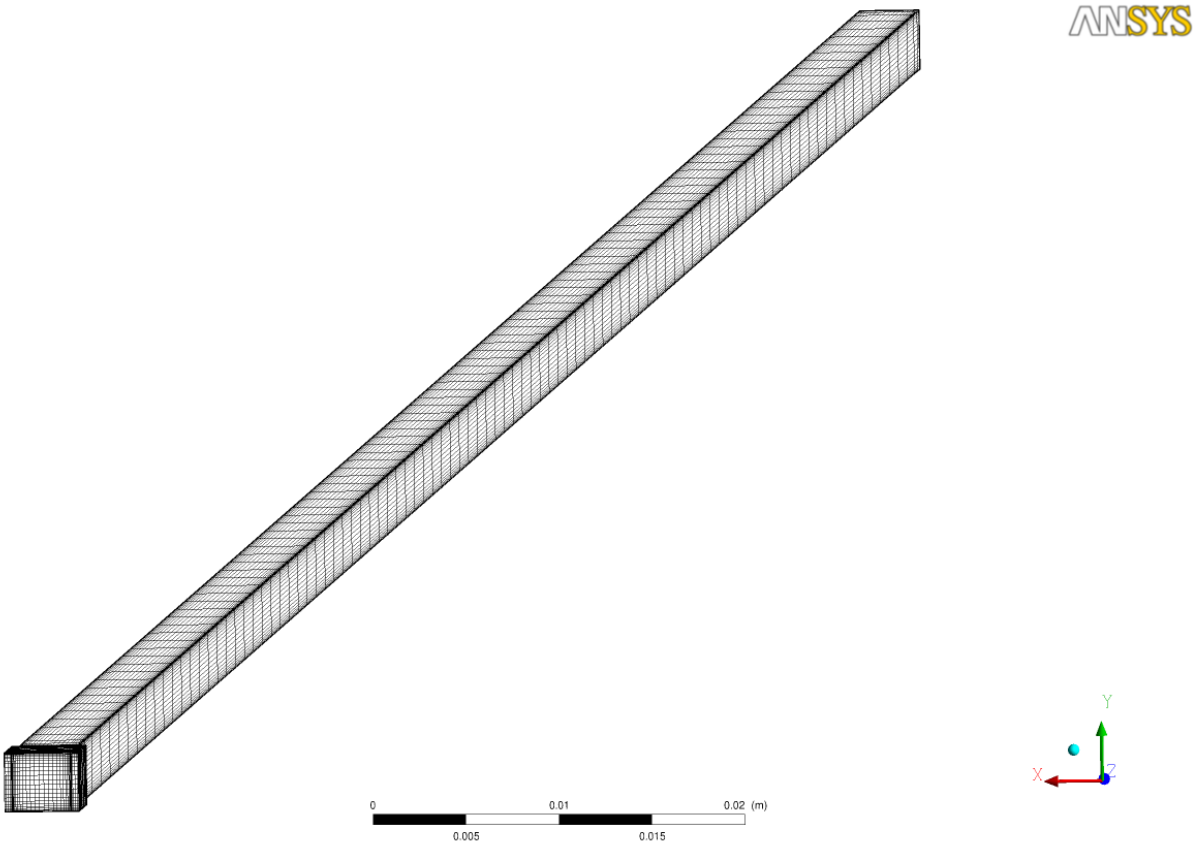


Figure 3.4.5 – *Fine mesh for one of the four tubes*

Table 3.4.3 – Mesh statistics for the fine mesh

Number of nodes:	144640
Number of elements:	134664
	Hexahedra elements
Total Volume:	$1.03616 \cdot 10^{-5}[m^3]$

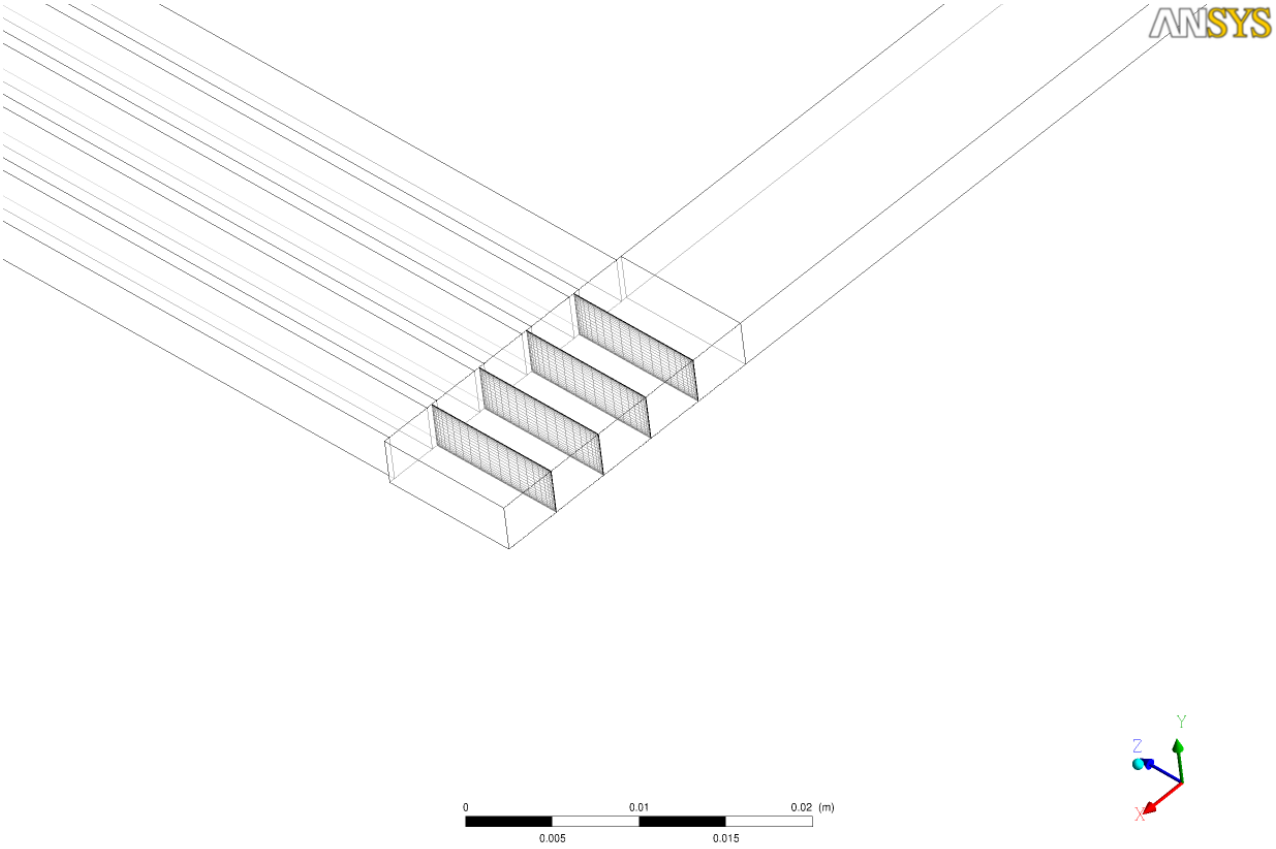


Figure 3.4.6 – *Close up of the interface region closest to the inlet seen in figure 3.4.4*

Boundary Conditions

Table 3.4.4 – Boundary conditions for five tube with inlet manifold simulation, the mass flow should be $0.005241228[kg/s] \cdot 5$ because of flow going down in 5 tubes.

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.005241228[kg/s]$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall
Flamewall	Heatflux	$q_j = 2.654377[MW/m^2]$
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall

3.5 Outlet Manifold

After the simulation with only a inlet manifold it was clear that the subdomains could handle a bit more complex simulation. Therefore the outlet manifold where introduced and an outlet similar to the one in the inlet where placed at the bottom of the tubes, see figure 3.5.1.

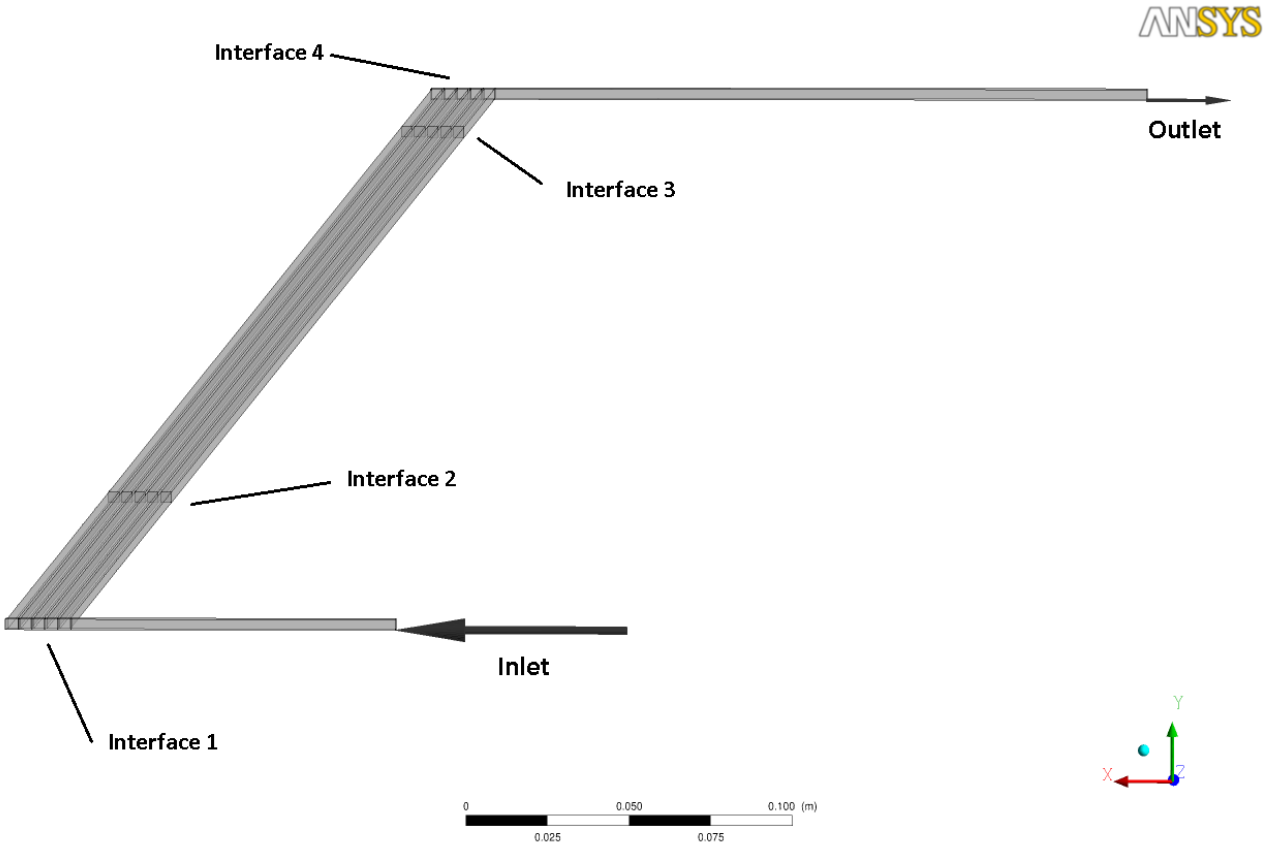


Figure 3.5.1 – *Domain for 5 tube with inlet manifold and outlet manifold simulation*

3.5.1 Mesh

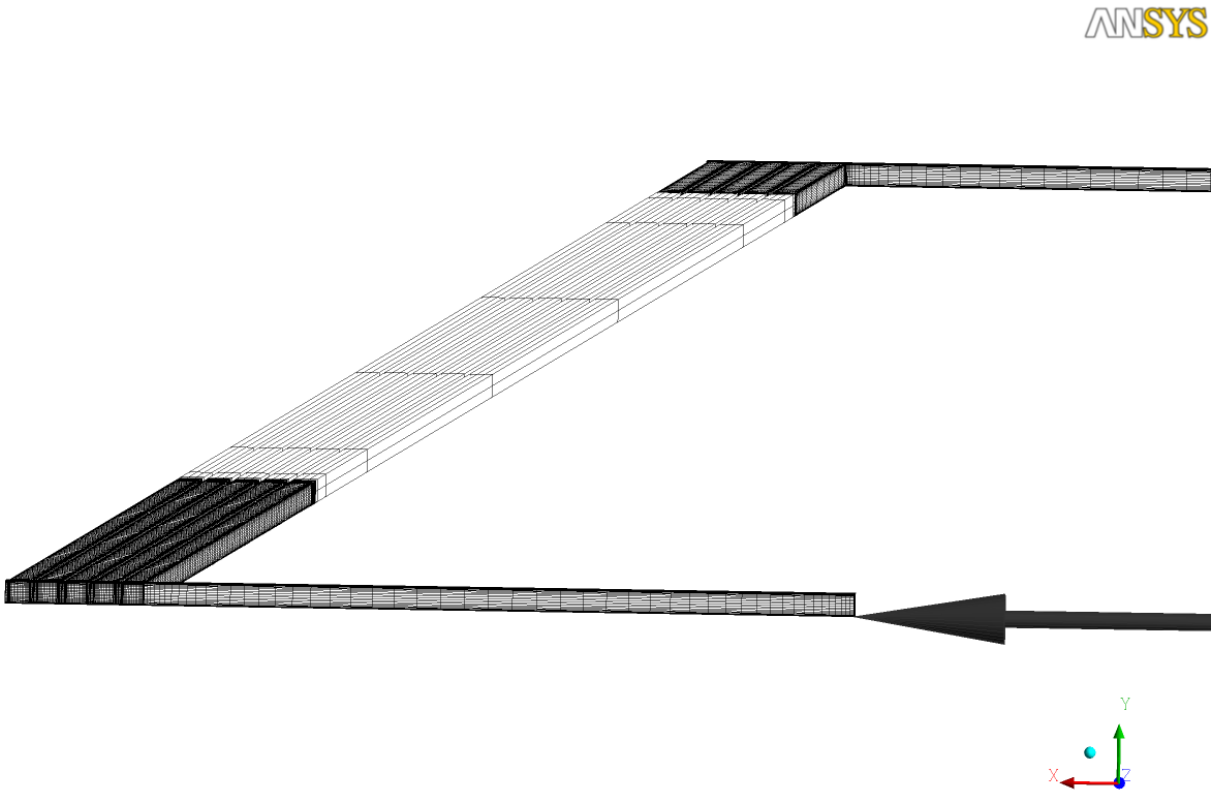


Figure 3.5.2 – *Mesh for simulation*

Table 3.5.1 – Mesh statistics for the simulation of both inlet - and outlet manifold

Number of nodes:	1197660
Number of elements:	1111760
	Hexahedra elements
Total Volume:	$2.25352 \cdot 10^{-4}[m^3]$

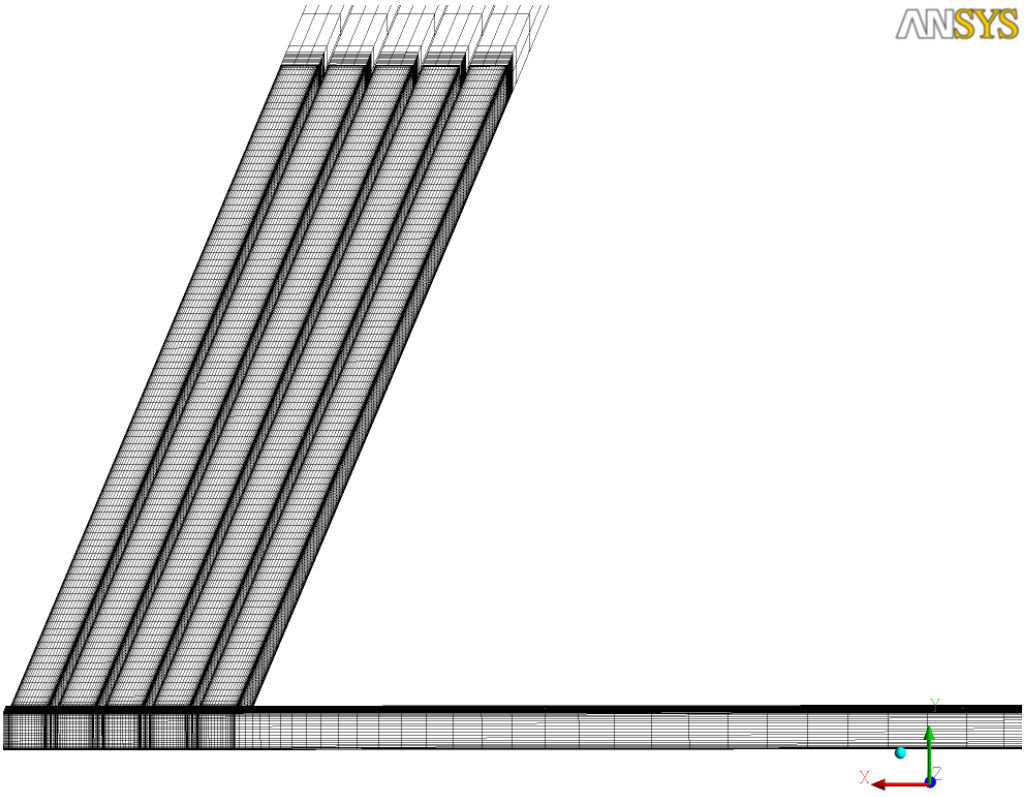


Figure 3.5.3 – *Zoom of the inlet region of the mesh figure 3.5.2*

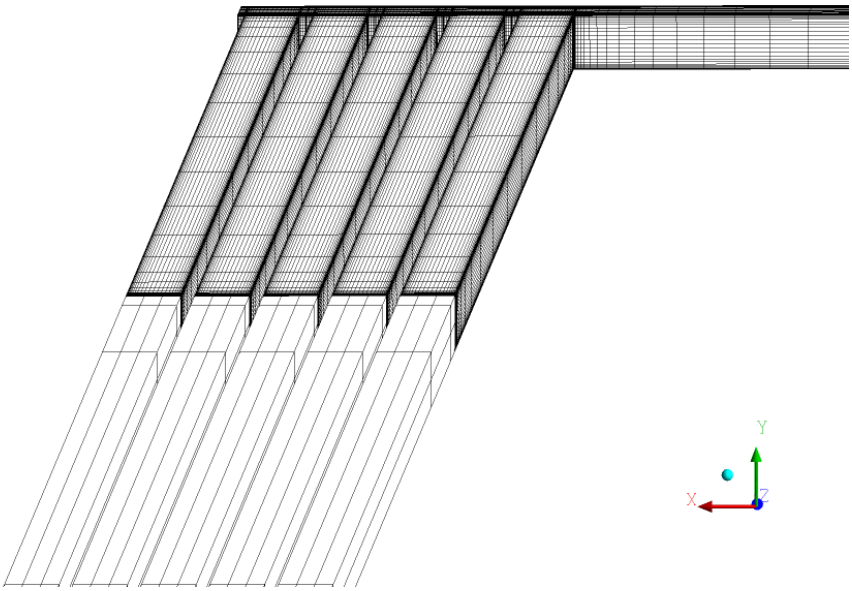


Figure 3.5.4 – *Zoom of the outlet region of the mesh figure 3.5.2*

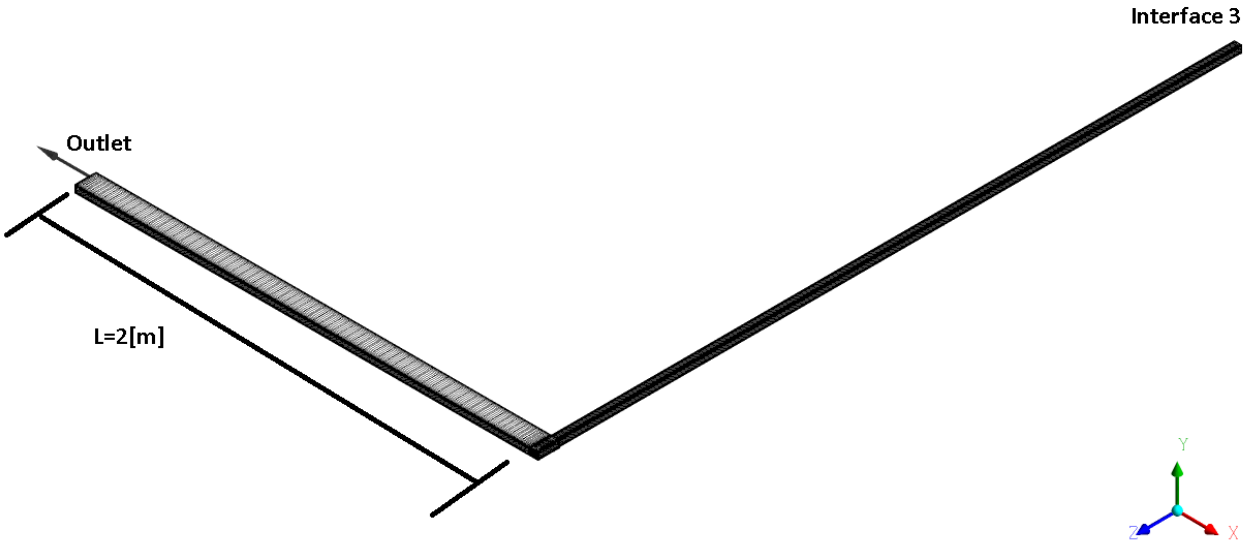


Figure 3.5.5 – *Zoom of the outlet tube in figure 3.5.2*

3.5.2 Boundary Conditions

Table 3.5.2 – Boundary conditions for interface investigation of the fine to coarse interface

Boundary Condition		
Boundary	Description	Value
Inlet	Mass flow	$0.0262061[kg/s]$ $(2.39[kg/s]/456 \cdot 5)$
	Static temperature	$40[K]$
Domain	Reference Pressure	$26[bar]$
Outlet	Static pressure	$0.7[bar]$
Walls	Adiabatic	
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall
Flamewall	Heatflux	$q_j = q(\dot{m})[MW/m^2]$
	Coarse region: Free slip	
	Fine region: No slip	Smooth wall

Varying heat flux, $q_j(\dot{m})$

The solid walls are not simulated and the temperature is forced to vary inside the channel. The heat flux is constant and this makes the system "stiff". When fully resolving the flow and looking at the heat transfer in the tube the simulation would apply a temperature on the flameside and then let the temperature convect through a solid wall with a heat transfer coefficient, h, then the amount of the cooling of the wall will depend of the mass flow in the tube. In the previous simulations the heat flux have not been varying and then the temperature of the wall *(imagining that there would be a wall in the simulation)* would only depend on the mass flow in the channels.

When the flow turns and goes down into the tubes the mass flow for each tube will vary compared to other tubes since the flow will experience a pressure gradient. The highest pressure will be found farthest to the left seen in figure 3.4.4 since here the flow will reach a stagnation point due to the "wall" boundary condition. When the mass flow vary the channel wall will be cooled different in the respective tubes. The heat load affecting the wall will increase if the mass flow increase and in order to take this into consideration the heat flux depend on the mass flow. To create this varying heat flux, nine simulations in $TCCOOL$ were preformed. In these simulations the mass flow varied from $1.912[kg/s]$ to $2.868[kg/s]$. The total heat flux from these simulations were plotted against the massflow specified for each case figure 3.5.6. As can be seen in figure 3.5.6 the heat flux were fitted with a third order polynomial and from this equation the mass flow were extended to give a broader span of mass flow, for one tube, from $0[kg/s]$ to $0.0085[kg/s]$. Naturally the heat flux is zero when the mass flow is zero, but this is simply to have an value to use in the simulation.

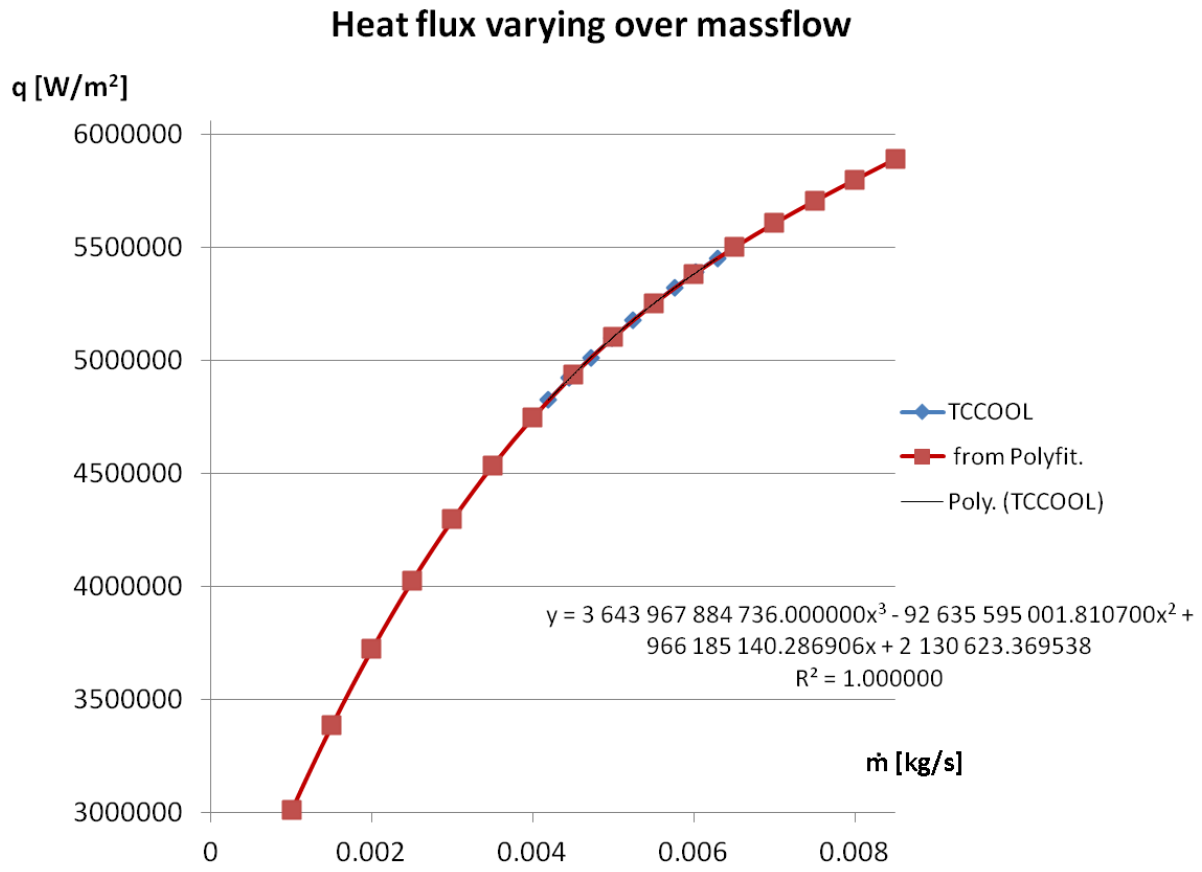


Figure 3.5.6 – *Heat flux depending of mass flow with polynomial curve fitting of TCCOOL extended over a greater variation.*

4 Results

In the following chapter the results will be presented in the following order:

- Solution details
- General results
- Some more specific results

4.1 Validation

The validation cases were mainly two different simulations. One with the heat flux slightly lower than the other. In order to see how low mesh resolution was possible to have also a section with the mesh dependence is presented, chapter 4.1.1. When it was clear that the subdomains could handle a more complex domain the validation case were re-run with a varying k_{loss} which is presented under the chapter 4.1.3.

4.1.1 Mesh dependence

This simulations boundary conditions is found in chapter 3.2.3. From figure 4.1.1 the pressure would not correlate well with the *TCCOOL* simulation. Only the simulation with the finest mesh, **Mesh 7**, resolution came close enought to the *TCCOOL* simulation to be considered to further improve the mesh in the following simulations.

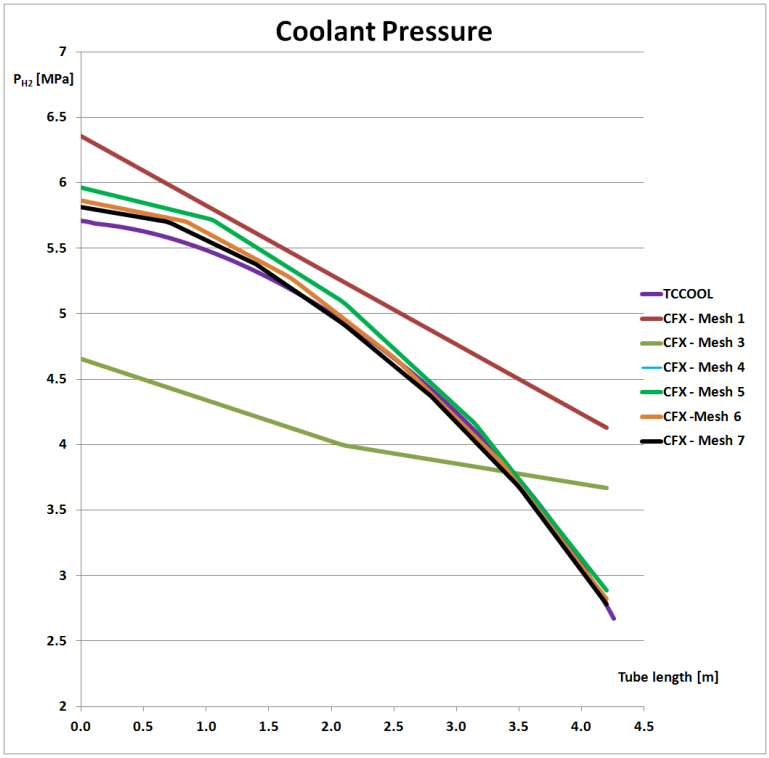
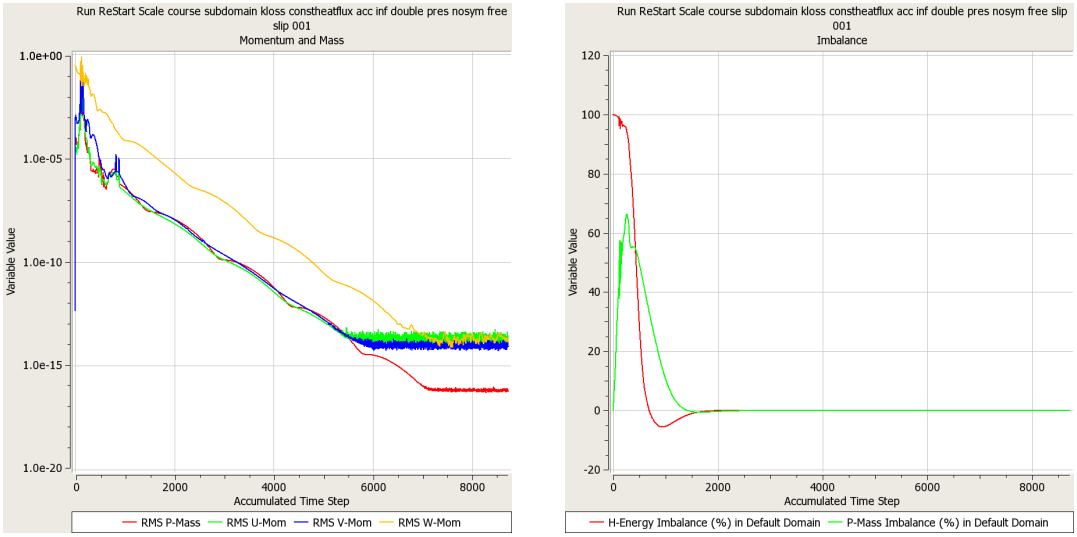


Figure 4.1.1 – *Pressure plot for the different meshes compared to the TCCOOL simulation*

4.1.2 Validation case 1

The simulation converged after 8718 iterations with a time step of $5.0 \cdot 10^{-5}[s]$. For a parallel run on the Volvo Aero cluster this simulation took, in total $1161.0[s]$, only CPU time required, $868.1[s]$. Figure 4.1.2 show the RMS residuals. When these residuals leveled out then the energy imbalance were looked at, seen in figure 4.1.2, the solution is in balance and can therefore be considered converged. As can be seen the imbalance in figure 4.1.2 were converged after 3000 iterations and the simulation could have been stopped earlier. Therefore it is assumed that this simulation could take a bit shorter time if a better convergence criteria, for example a limit of the derivative for the energy imbalance, could be set. When the simulation were stopped the imbalance were below $1 \cdot 10^{-11}$ and had been below $1.2 \cdot 10^{-8}$ for the last 2000 iterations.



(a) **(b)**

Figure 4.1.2 – *Solver: a)RMS residuals for mass- and momentum equation and b)Imbalance plot of the solution for energy equation*

The results show that when k_{loss} is $9.14[m^{-1}]$ the temperature averaged over the outlet of the domain T_{out} is $898[K]$ and the averaged pressure P_{in} over the inlet is $55.67[bar]$, which were considered to be sufficiently close to the simulations results from *TCCOOL*. In all further simulations, this k_{loss} value will be used. Figure 4.1.3 show the temperature in the tube compared with the *TCCOOL* simulation. Figure 4.1.4 show the pressure in the tube compared with the *TCCOOL* simulation. The coarse mesh can follow the *TCCOOL* simulation quite well. The behavior of the pressure is captured quite well and the temperature are matched in the inlet and outlet but does not follow the exact same behavior as the *TCCOOL* simulation. This is due to the constant heat flux used in *CFX* which differs from the varying heat flux, $q_j(z)$ in figure 3.2.4, in *TCCOOL*.

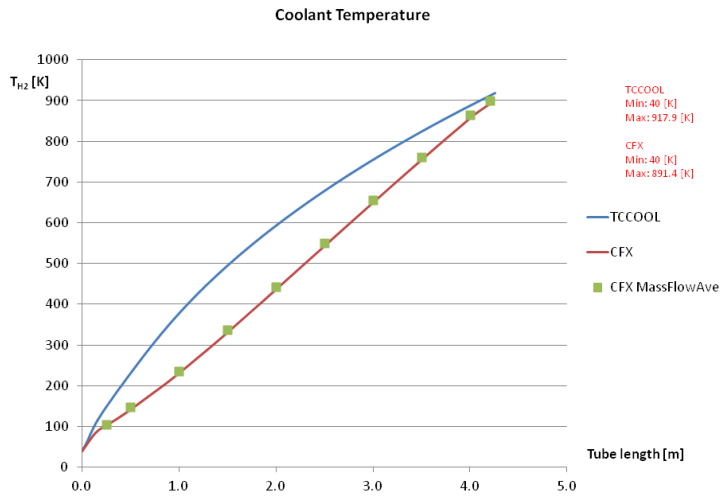


Figure 4.1.3 – *Temperature plot for the center of the channel (red curve) and average over cross section, plane in XY, (green squares) of the CFX simulation compared to the TCCOOL simulation (blue curve).*

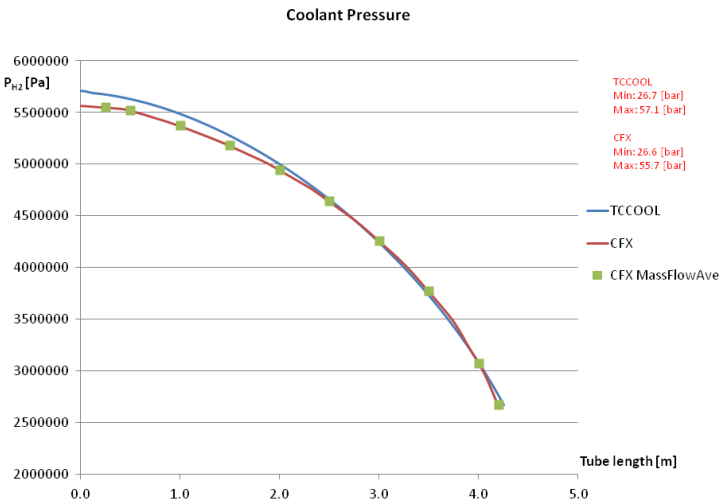


Figure 4.1.4 – *Pressure plot for the center of the channel (red curve) and average over cross section, plane in XY, (green squares) of the CFX simulation compared to the TCCOOL simulation (blue curve*

.

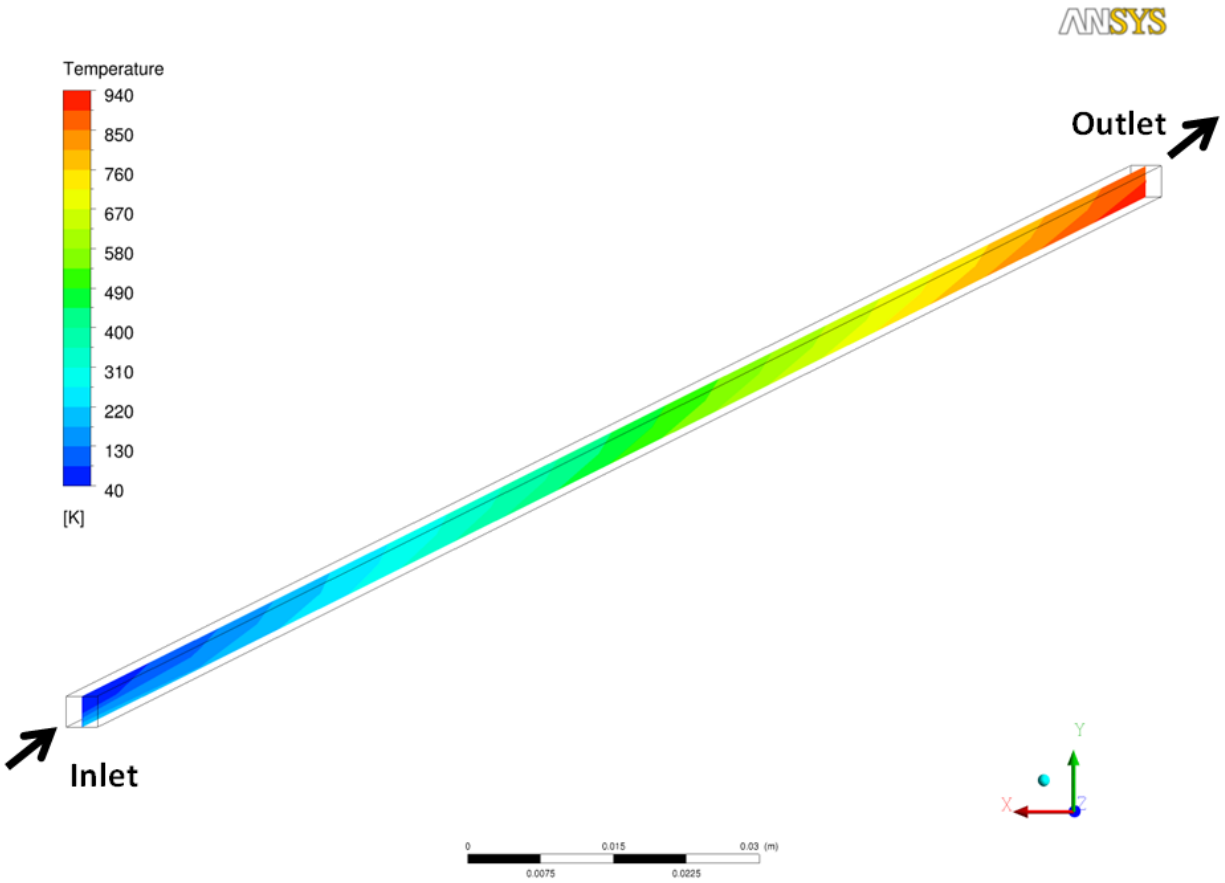


Figure 4.1.5 – *Contour plot of temperature at YZ-plane at X = 1.6 [mm] (center of the channel), inlet is seen here where the coolant is the coldest, in the left of the picture*

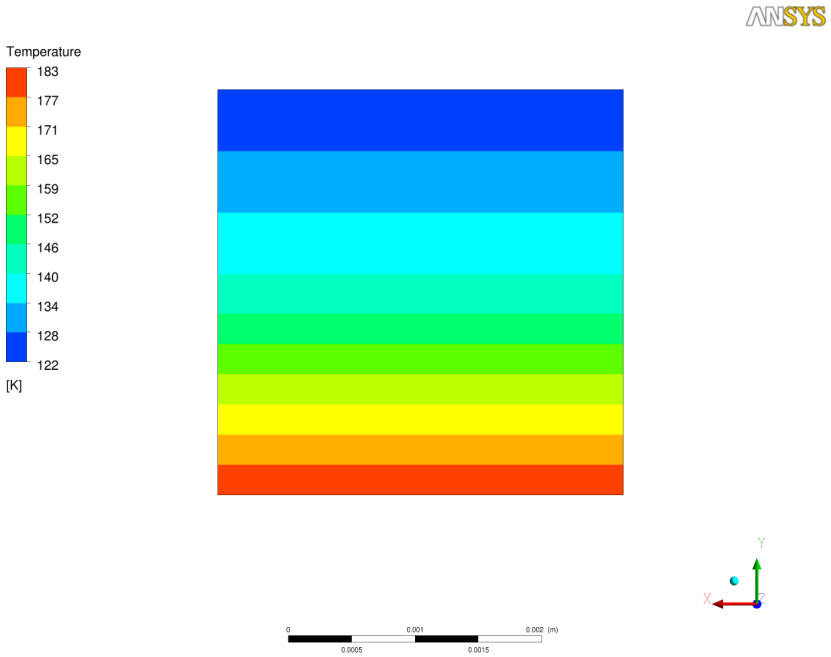


Figure 4.1.6 – *Contour plot of temperature at XY-plane Z = 0.5 [m]*

The heat flux flows into the domain through the flame wall seen from figure 4.1.5 and figure 4.1.6 also a stratification can be seen in figure 4.1.6. But seen in figure 4.1.5 this stratification has little influence over the cross section of the channel and can therefore be considered almost constant. From the results it is possible to calculate the heat flux that goes in to the domain through the flamewall via the area integral of the flamewall for the heat flux:

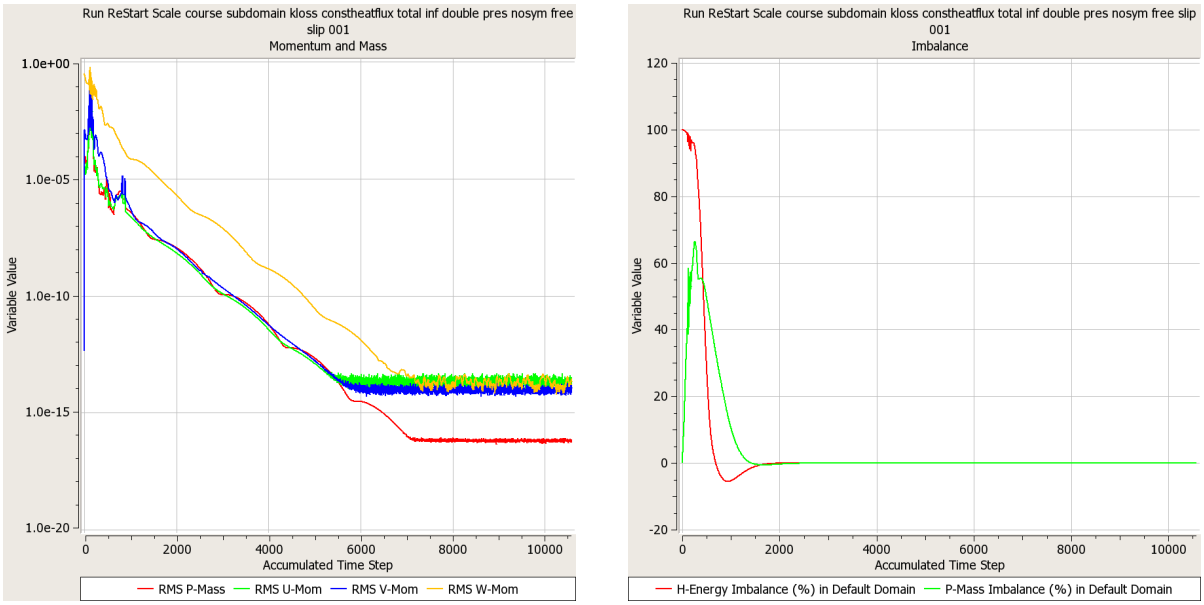
$$Q = \int_{flamewall} q_j dA = 69.2099 \approx 69.21[kW]$$

Which also can be verified by the heat flux, prescribed in the boundary conditions, see table 3.2.4, $q_j = 5.1495[MW/m^2]$, where:

$$Q = A_{flame\,wall} \cdot q_j = 0.01344 \cdot 5.14954 \cdot 10^6 = 69.21[kW]$$

4.1.3 Validation case 2

The simulation converged after 10579 iterations with a time step of $5.0 \cdot 10^{-5}$. For a parallel run on the Volvo Aero cluster this simulation took, in total 1213.0[s], the CPU time, which is the time required for only solving the equations removing the time that it takes to write the solution, was 934.7[s] which is about 15[min]. Figure 4.1.7 show the RMS residuals. When these residuals leveled out then the energy imbalance were looked at. Seen in figure 4.1.7 the solution are in balance and can therefore be considered to be converged. When the solution were stopped the imbalance where below $5 \cdot 10^{-12}$ and had for the last 2000 iterations been below $1.5 \cdot 10^{-11}$. This is very low and as can bee seen in the following simulations the imbalance is allowed to be considered converged significantly higher. With the slightly higher heat flux, $q_j = 5.174409[MW/m^2]$, the temperature

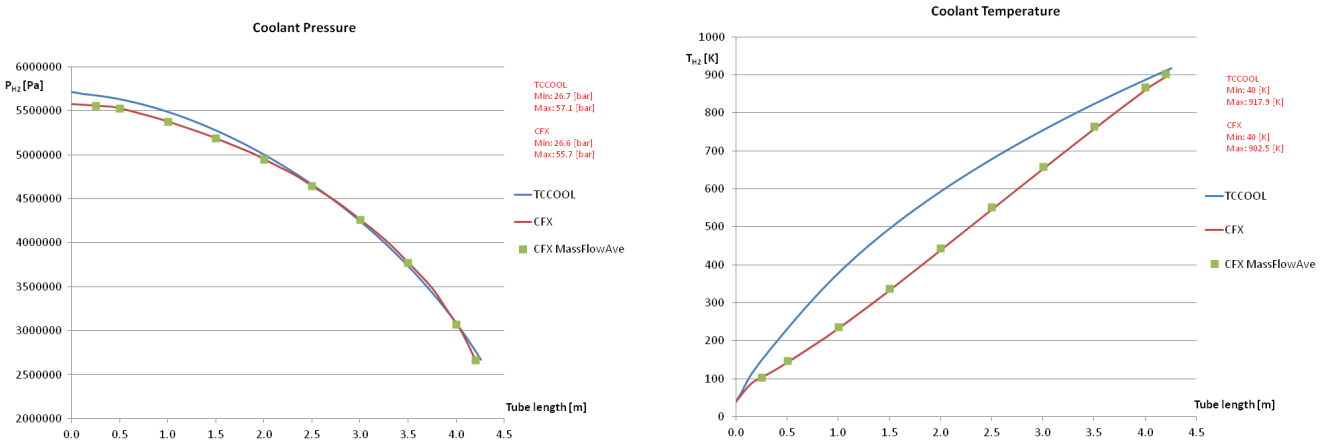


(a) **(b)**

Figure 4.1.7 – *Solver: a)RMS residuals for mass- and momentum equation and b)Imbalance plot of the solution for energy equation*

increased, $T_{out} = 902.5[K]$ and also pressure increases slightly, $P_{in} = 55.77[bar]$. This result show that when a slightly higher heat flux is applied the solution in CFX is closer to the $TCCOOL$ and it can therefore be assumed that the slightly higher heat flux is the more correct. As expected, the highest pressure is found at the inlet and also here the temperature is lowest, seen in figure 4.1.8. Further downstream, the temperature increase, due to the temperature gradient in the heat flux in equation 2.2.12. Since the temperature increase the velocity increase, seen in figure 4.1.9, the pressure decrease.

By taking the two results files into post there is a possibility to subtract the results from the second simulation from the results from the first and this is what can be seen in figure 4.1.10. The difference in heat flux affects the temperature the most in the outlet, meaning that the total heat taken up by the fluid is less when a lower heat flux is applied, which is as expected.



(a) *Pressure* **(b)** *Temperature*

Figure 4.1.8 – *Plot for the center of the channel (red curve) and average over cross section, plane in XY-direction, (green squares) of the CFX simulation compared to the TCCOOL simulation*

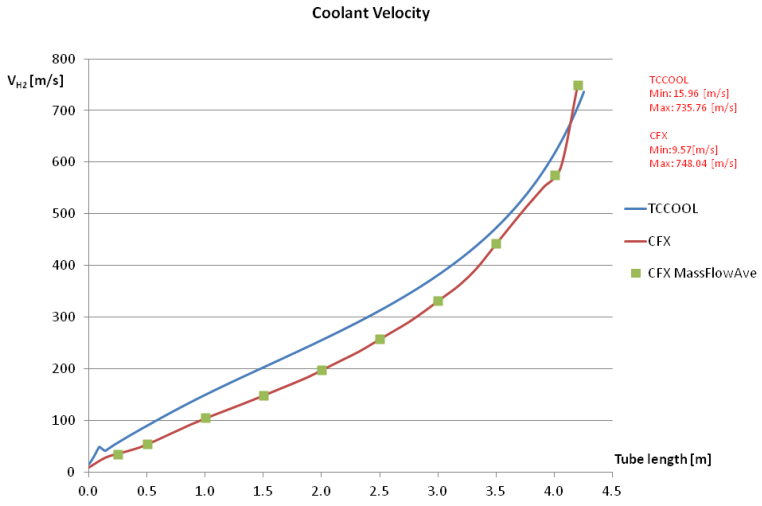
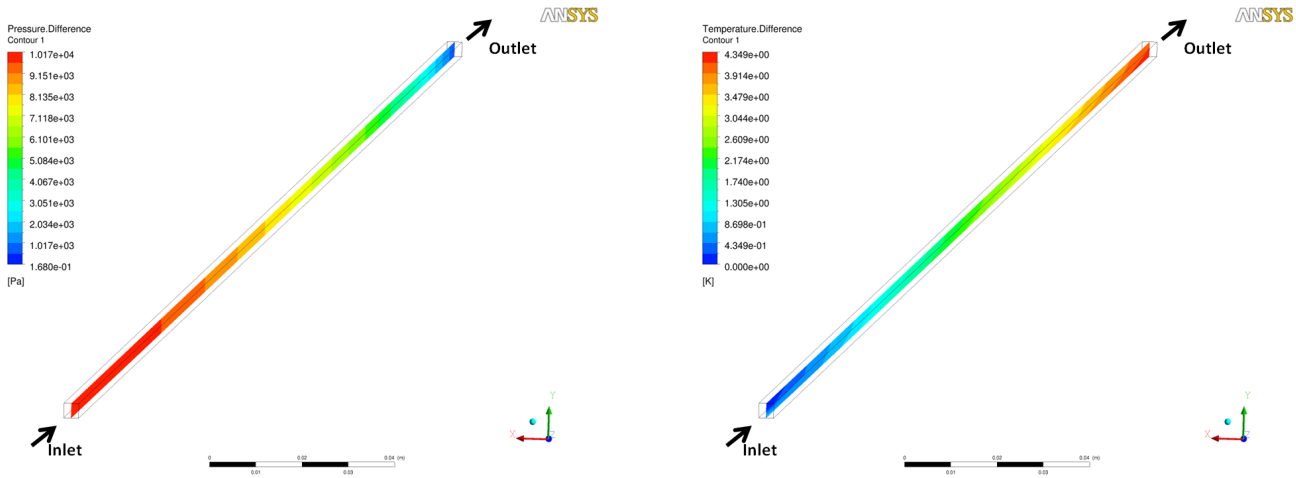


Figure 4.1.9 – *Comparison of velocity from TCCOOL simulation and CFX*

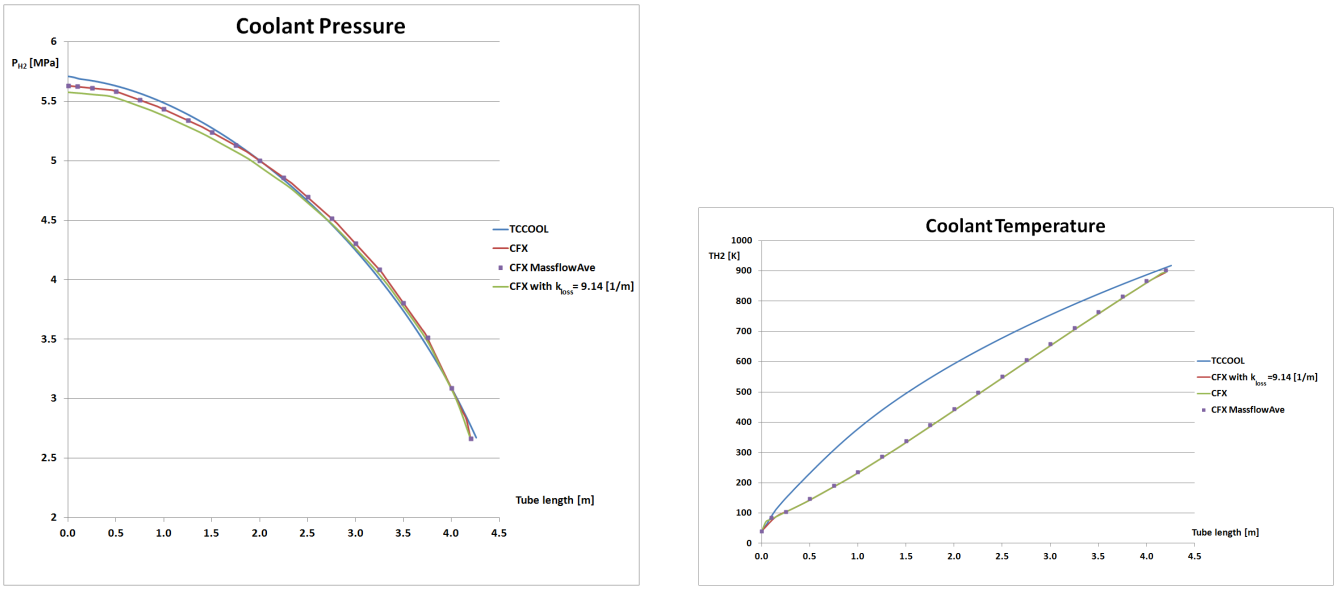


(a) *Pressure* **(b)** *Temperature*

Figure 4.1.10 – *Contour plot of difference between Validation case 1 and 2*

Varying k_{loss}

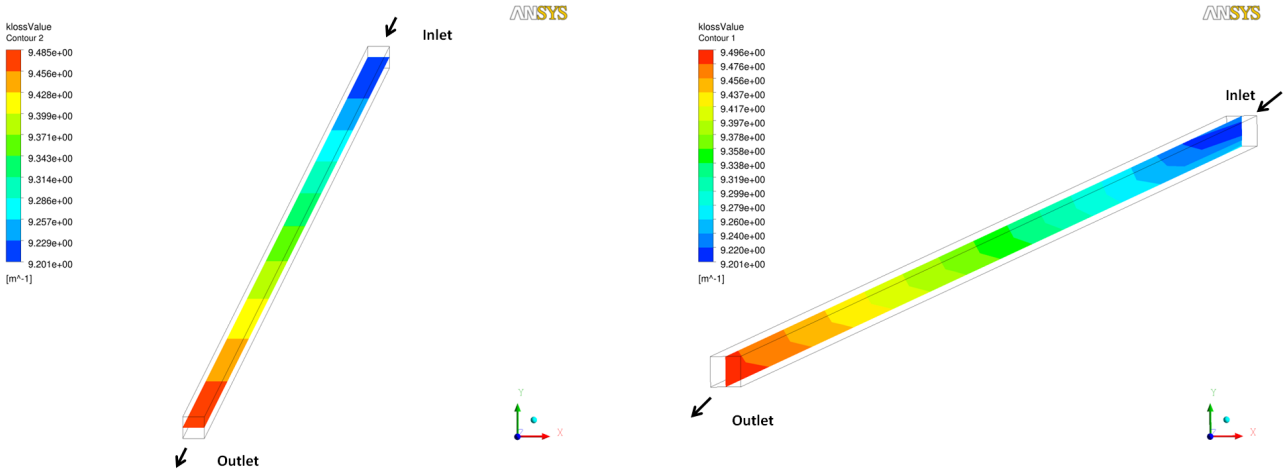
The *validation case 2* were also run with a varying k_{loss}, as explained in chapter 3.2.4. The pressure showed a better correlation with the *TCCOOL* simulation, as can be seen in figure 4.1.11. The pressure only differ -0.0728% between the *TCCOOL* simulation and the *CFX* simulation on the inlet compared with the difference of -2.508% in the simulation *Validation case 2* compared with the *TCCOOL* simulation. The difference in temperature on the outlet is only -2.456% between the *CFX* simulation with a varying k_{loss} and the *TCCOOL* simulation.



(a) *Pressure* **(b)** *Temperature*

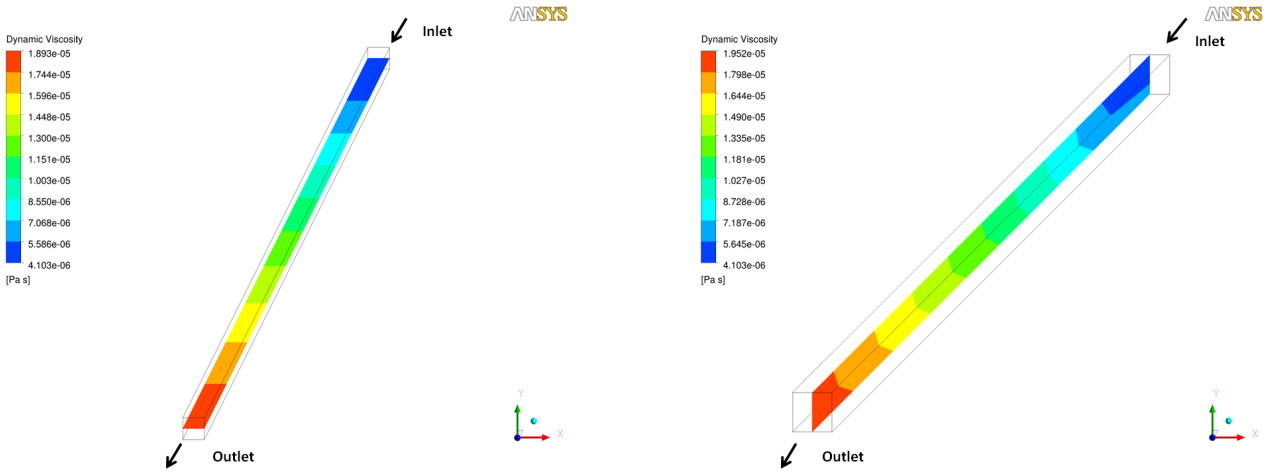
Figure 4.1.11 – *Plot for the center of the channel with varying k_{loss} (red curve), massflow average with varying k_{loss} (purple squares), the center of the channel with a constant k_{loss} (green curve), and TCCOOL simulation (blue curve)*

In figure 4.1.12 the variation of k_{loss} is seen. It is clear that k_{loss} actually is varying, according to the Serghides friction factor equation 2.1.9, since the Reynolds number is varying which in turn is varying because the dynamic viscosity is varying, seen in figure 4.1.13. The dynamic viscosity is varying according to the pressure-temperature dependence of the real gas. The Reynolds number is in fact varying in each cell but since the purpose is to compare it with *TCCOOL* which is one dimensional the variation should only be in streamwise direction, but this was not accomplished in this work.



(a) *XZ-plane in center of the tube in y-direction* **(b)** *YZ-plane in center of the tube in x-direction*

Figure 4.1.12 – *Contour plots of the variation of k_{loss} in the streamwise direction.*



(a) *XZ-plane in center of the tube in y-direction* **(b)** *YZ-plane in center of the tube in x-direction*

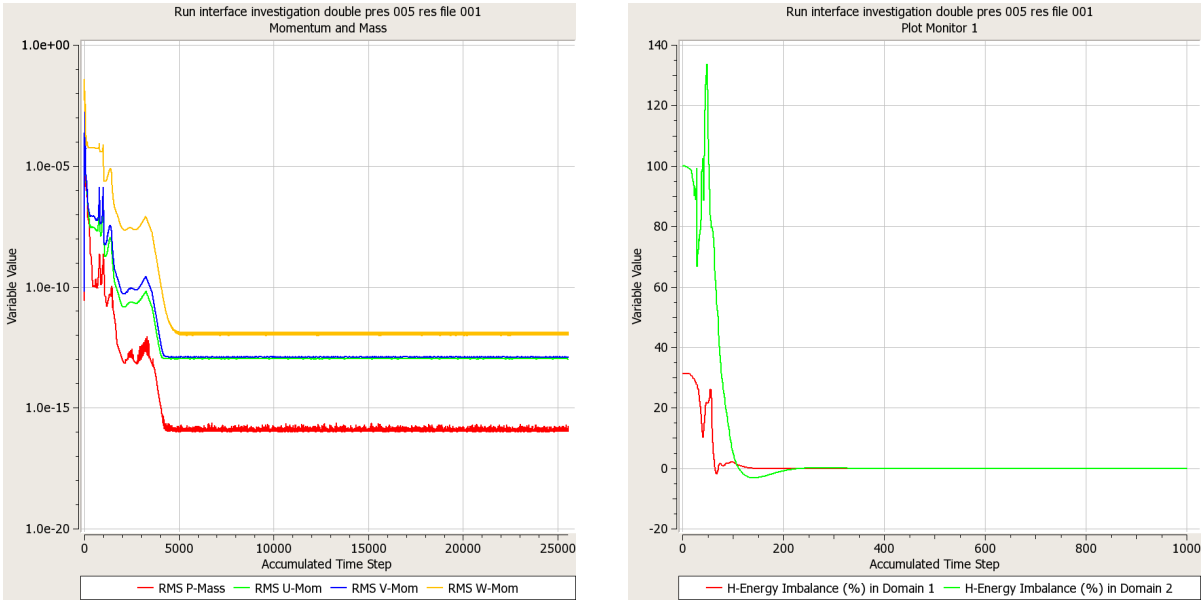
Figure 4.1.13 – *Contour plots of the variation of dynamic viscosity in the streamwise direction.*

4.2 Interface Investigation

In this chapter the results from the cases explained in the method chapter under chapter 3.3 are presented. First the case where the mesh goes from a fine mesh region to a coarse mesh region, chapter 4.2.1, is presented. Secondly the case where the mesh goes from a fine mesh region to a coarse mesh region again to a fine mesh region, chapter 4.2.2, is presented. These two simulations are compared, found under chapter 4.2.3. Since these results were found to be influenced by the mesh expansion factor the final results presented in this chapter is about the mesh expansion factor influence, chapter 4.2.4.

4.2.1 Fine to coarse simulation

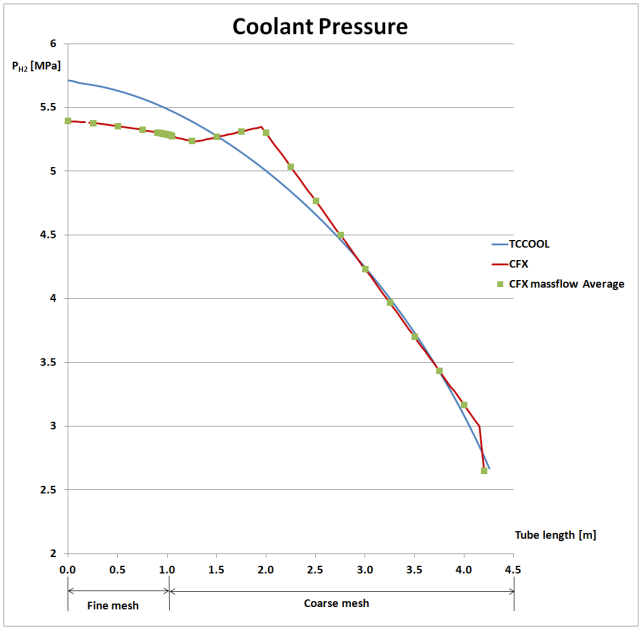
The solution converged after 25551 iterations with a time scale of $5.0 \cdot 10^{-4}[s]$. Clearly seen in figure 4.2.1, the simulations could be stopped already at 5000 iterations but the convergence criteria include that the imbalance of the energy equation, figure 4.2.1, went to zero also this criteria needed to be satisfied, which it was already at 400 iterations. When the solution were stopped the imbalance were below $1.5 \cdot 10^{-10}$ and for the last 20000 iterations the imbalance had been fluctuating between $2 \cdot 10^{-10}$ and $-3 \cdot 10^{-10}$. The simulation could have been stopped at ca 5000 iterations.



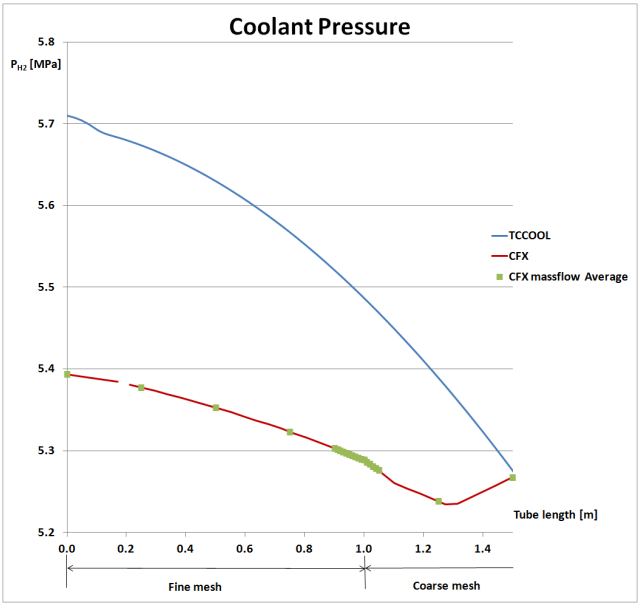
(a) **(b)**

Figure 4.2.1 – *Solver: a)RMS residuals for mass- and momentum equation and b)Imbalance plot of the solution for energy equation for first 1000 iterations*

Pressure	Inlet	$53.9 \cdot 10^6[Pa]$
	Outlet	$26.5 \cdot 10^5[Pa]$
Temperature	Inlet	$40[K]$
	Outlet	$894.3[K]$



(a) *Full tube*



(b) *Zoom of interface region*

Figure 4.2.2 – *Pressure plot over the centerline of the tube and mass flow averaged value compared with TCCOOL simulation, chapter 3.2.1*

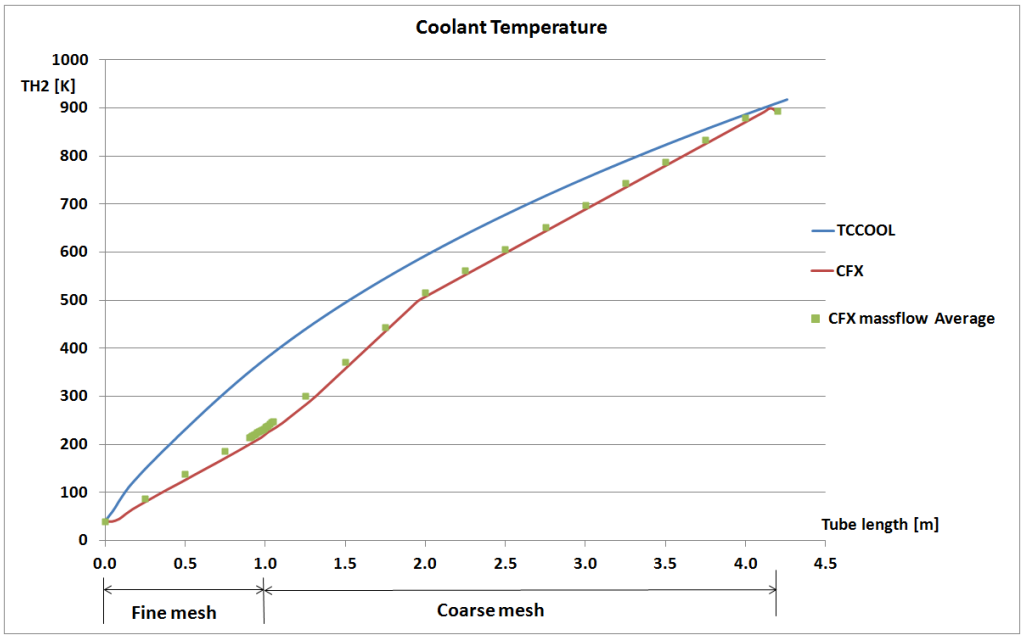
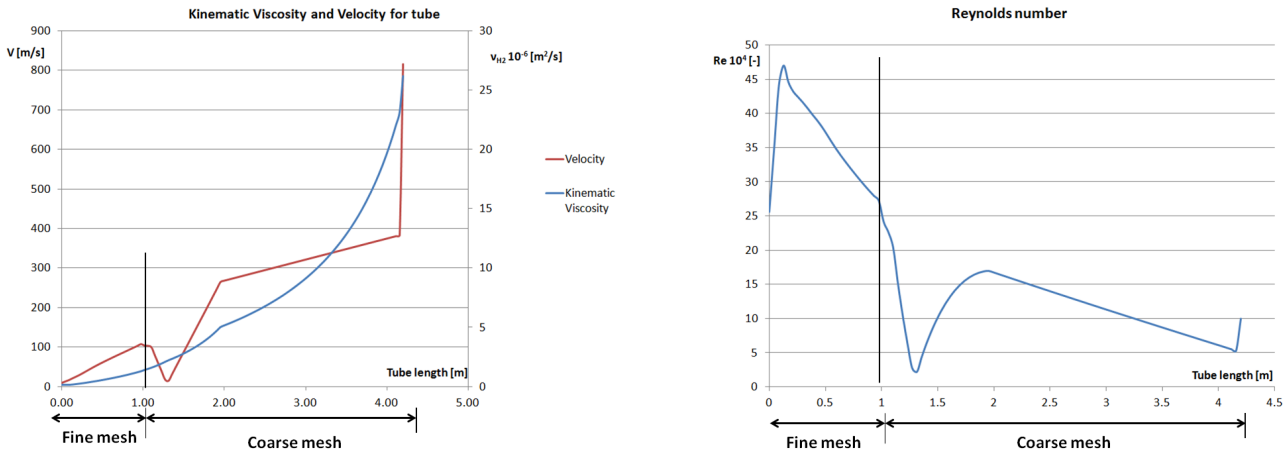


Figure 4.2.3 – *Temperature plot for TCCOOL simulation together with CFX simulation both centerline and mass flow averaged values, chapter 3.2.1*

The results were compared with the TCCOOL simulation from chapter 3.2.1. The velocity is affected the most by the interface and does not recover completely until the outlet. Also seen is that all values in the fine region is generally a bit low. This is due to that the boundary condition for the roughness of the walls are set to smooth, meaning that CFX treats the wall as hydraulically smooth. Hydraulically smooth mean that the wall roughness is less than 1/2 of the viscous sub layer height. This assumption may have some affect to the pressure and velocity discrepancies in the result for the simulation. The Reynolds number, figure 4.2.4, can be found by looking at the center of the channel. From the centerline the velocity and kinematic viscosity is plotted, figure 4.2.4. The interface is placed at $1[m]$ and it is right after the drop in velocity occurs, seen in figure 4.2.5. This discrepancy can also be seen in the pressure, see figure 4.2.2, although not as large. The reason for this discrepancy can be found in the mesh, the expansion factor is too large where the coarse mesh goes from the refinement near the interface and to the next cell. This problem is also seen in the end of the tube where the coarse mesh goes from the really coarse cells to a refinement towards the outlet. From the result file, .out-file, from *CFX* the mesh expansion factor is found. For this simulation it was 5979 but should have been below 20, according to [2]. For a perfect mesh the maximum mesh expansion factor should be below 5, acceptable between $5.0 - 20.0$ and poor status for > 20.0. In the chapter 4.2.4 the influence of this expansion factor were investigated.



(a) *Velocity and Kinematic viscosity* **(b)** *Reynolds number*

Figure 4.2.4 – *Plot over the centerline of the tube compared with TCCOOL simulation, chapter 3.2.1.*

The normalized wall distance for the fine mesh used in this case were well below the required value of $y^+ < 1$. On the **flamewall** the normalized wall distance were $0.05 < y^+ < 8.7$, as can be seen in figure 4.2.6. For the surrounding walls the normalized wall distance were $0 < y^+ < 16.3$, this is acceptable since it is assumed that the automatic wall function will handle this slightly higher normalized wall distance and no heat transfer occur on these walls. As the velocity increase the wall shear stress increase, since a larger velocity creates a larger velocity gradient normal to the wall, and also the y^+ increase since the y^+ depend on the wall shear stress as in equation 2.3.30.

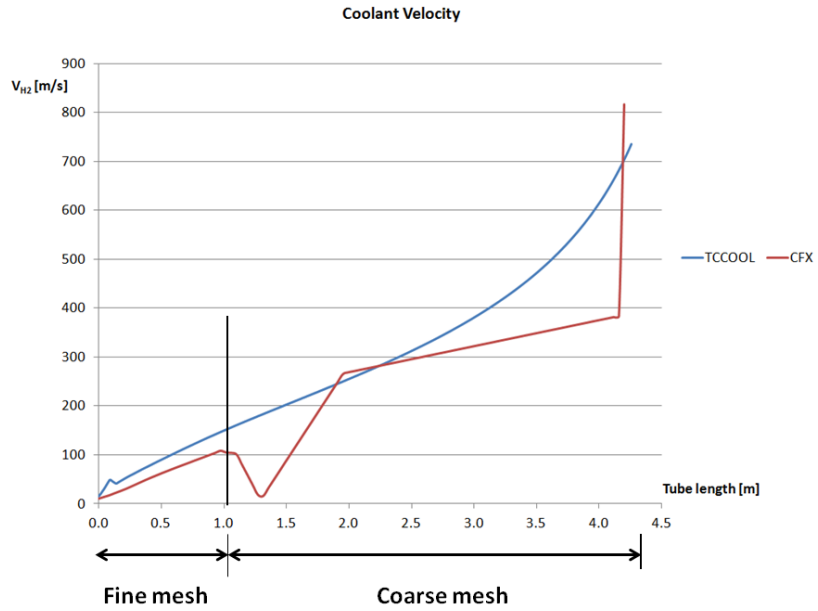


Figure 4.2.5 – *Comparison of velocity from TCCOOL simulation and CFX*

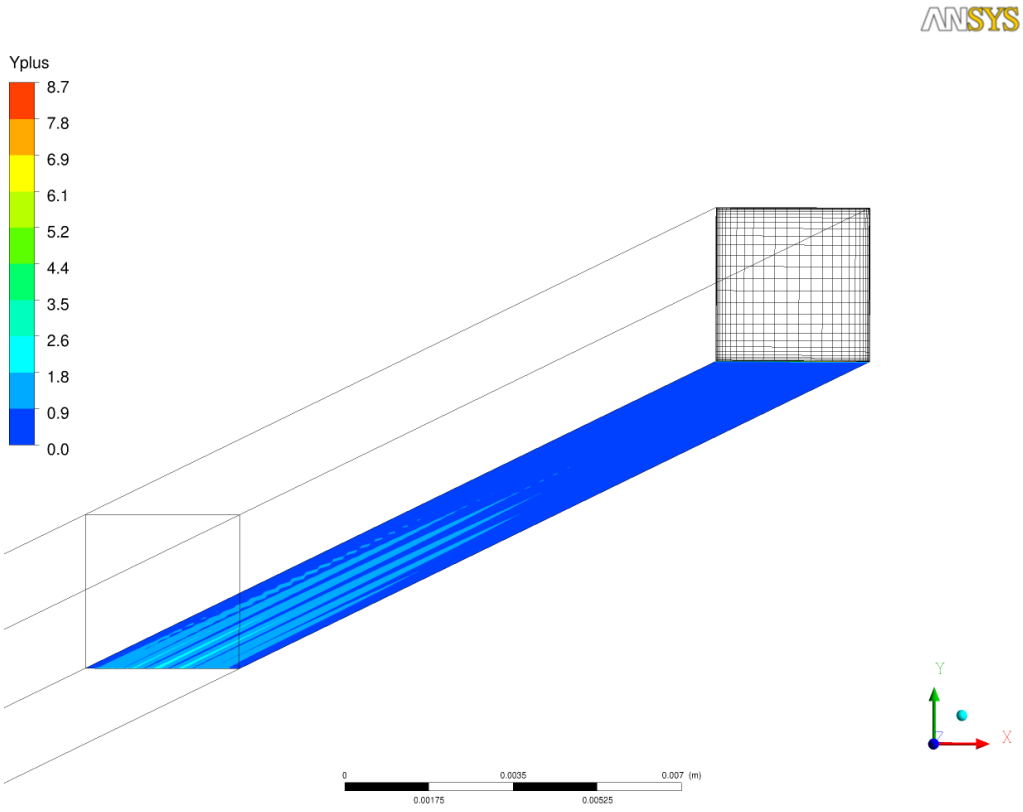
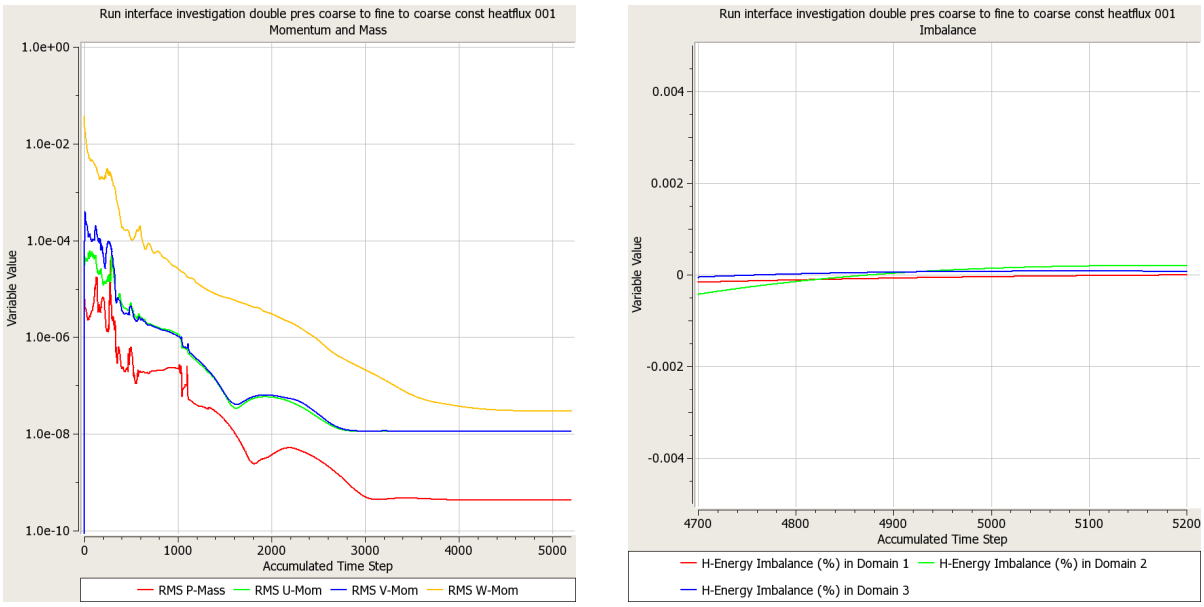


Figure 4.2.6 – *Contour plot colored with normalized wall distance, y^+, on flamewall of the fine mesh near the inlet. The surface of the mesh on the inlet is showed.*

4.2.2 Fine to coarse to fine simulation

The solution converged after 5200 iterations with a time scale of $5.0 \cdot 10^{-5}[s]$. The energy imbalance were for *domain 1* $-3.25 \cdot 10^{-6}$, for *domain 2* $2.0604 \cdot 10^{-4}$ and for *domain 3* $7.5 \cdot 10^{-5}$. The domains are numbered in order starting with the first fine region according to figure 3.3.6. For the last 1000 iterations the imbalance were below $0.5 \cdot 10^{-3}$ and above $-0.4 \cdot 10^{-2}$. The results are compared to the *TCCOOL* simulation and the simulation with only one interface in the next chapter, see chapter 4.2.3.

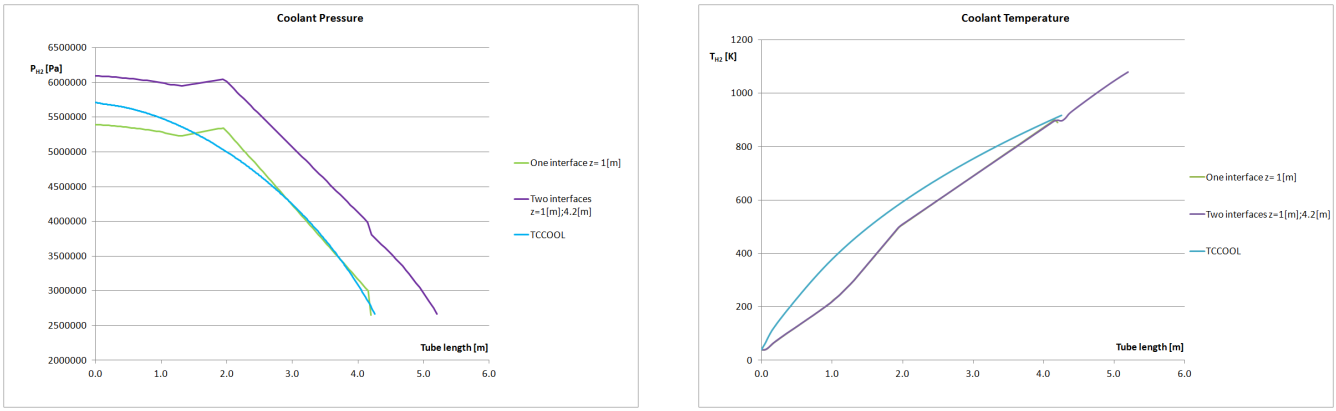


(a) **(b)**

Figure 4.2.7 – *Solver: a)RMS residuals for mass- and momentum equation and b)Imbalance plot of the solution for energy equation.*

4.2.3 Comparison

When comparing the two *CFX* simulations together with the *TCCOOL* simulation the differences are clearly seen. The pressure is the most affected and in the two interface simulation the pressure is about 13-32 % higher than with one interface. But since the simulation with two interfaces use the mesh from the one interface for the fine region this domain is actually 1[m] longer than the simulation with only one interface. Therefore the pressure decrease over the channel is bigger, and since the pressure is set on the outlet the inlet pressure will be higher. As seen in simulation with only one interface the pressure and velocity is the most disturbed of the interfaces. In average is the pressure varying 17.5 % and the velocity -11.8 % compared to the temperature which in average is varying 0.0%. The percentage averaged values are created by subtracting the *CFX* result from the *TCCOOL* result, dividing by the *TCCOOL* value and then summing them up and dividing by the number of values.



(a) *Pressure* **(b)** *Temperature*

Figure 4.2.8 – *Plot over the centerline of the tube compared with TCCOOL simulation, chapter 3.2.1*

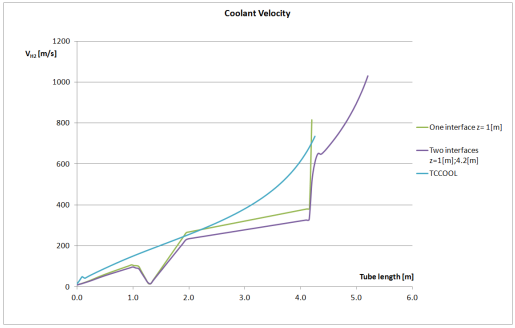


Figure 4.2.9 – *Comparison of velocity from TCCOOL simulation and CFX plotted over the centerline of the tube*

4.2.4 Expansion factor influence

The expansion factor were below 20 when the meshing parameters were set to the values in table 4.2.1 for the long edges of the mesh in figure 3.3.3. The new mesh can be seen in figure 4.2.11, it is a smoother transition from the finer mesh to the coarser mesh, and also from the coarse mesh to the finer region, than the previous mesh used in the simulations.

The coarse mesh in simulation described in 3.3.1 were replaced with the mesh described above and re-run on the Volvo Aero cluster. The convergence criteria were the same as previous simulation. As can be seen in

Table 4.2.1 – Mesh parameters, for explanation see Appendix B

Nodes	10
Mesh law	Biexponential
Spacing 1	$1[mm]$
Ratio 1	2
Spacing 2	$1[mm]$
Ratio 2	2
Max space	$1 \cdot 10^{10} = 1863.01[mm]$

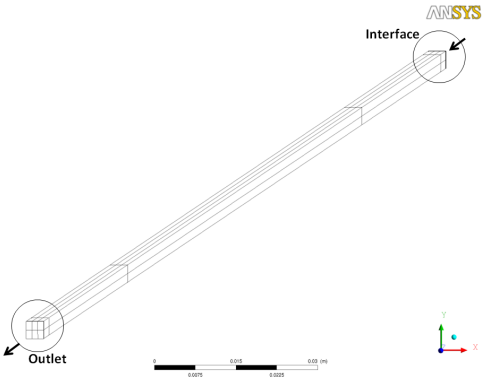
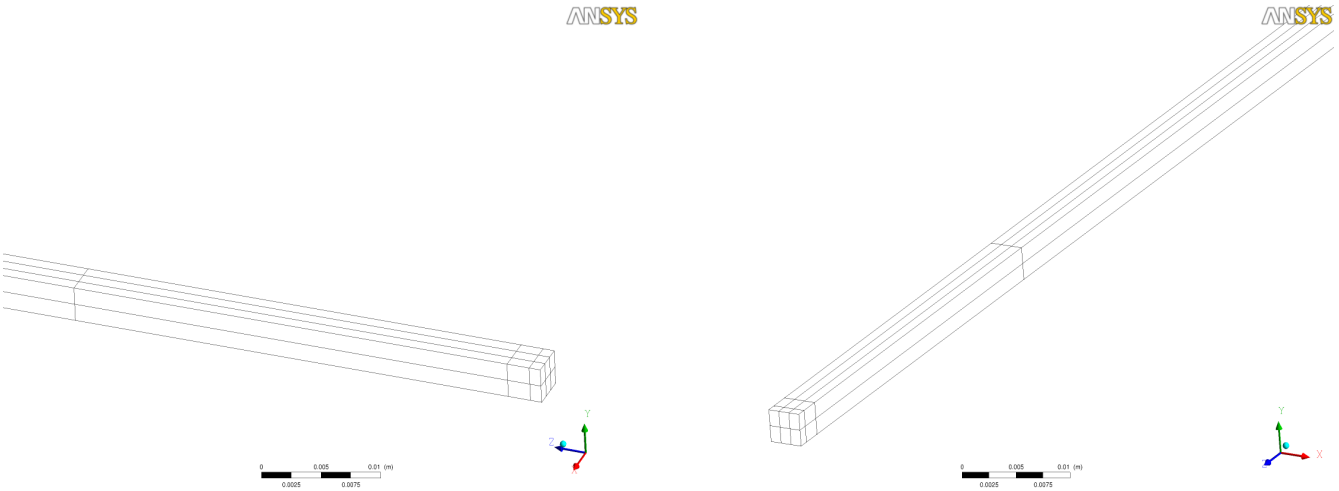


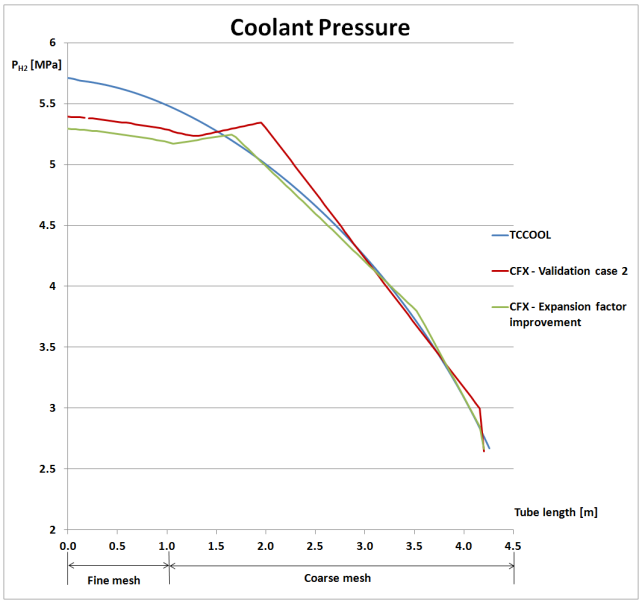
Figure 4.2.10 – *Mesh for investigation of the mesh expansion factor influence*



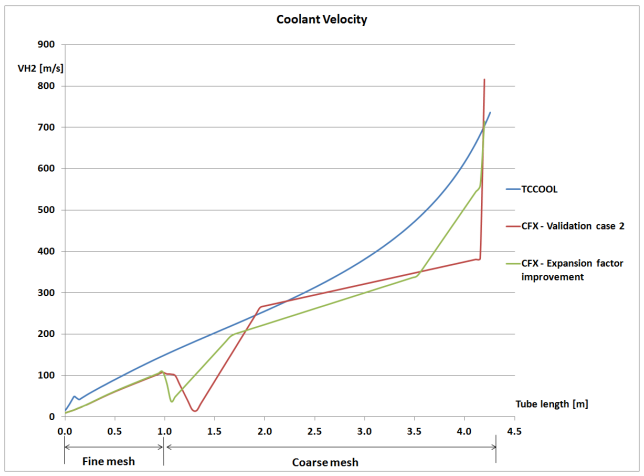
(a) *Upper right corner* **(b)** *Lower left corner*

Figure 4.2.11 – *Zoom of the region with a circle around it in figure 4.2.11.*

figure 4.2.12 the influence of the expansion factor is clearly seen. When the expansion factor were decreased the pressure came closer to the *TCCOOL* simulation after the interface, which was expected. Also the temperature is affected by the expansion factor, as seen in figure 4.2.13.



(a) *Pressure*



(b) *Velocity*

Figure 4.2.12 – *Plot of the center of the tube compared with the TCCOOL simulation with a expansion factor* < 20 *of the mesh*

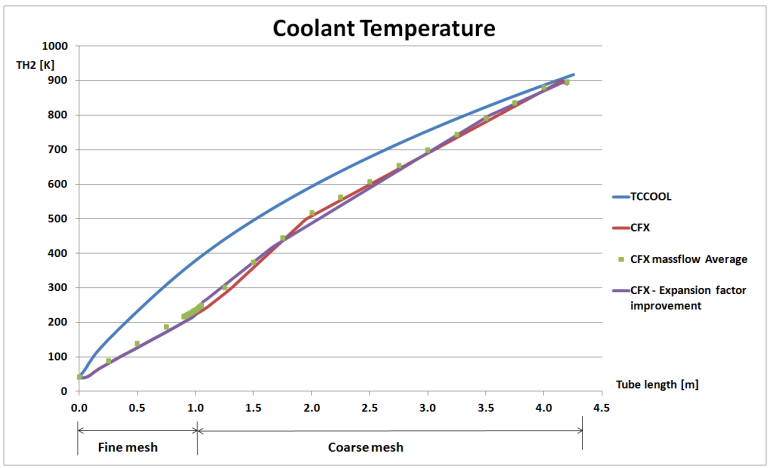


Figure 4.2.13 – *Plot of the temperature in the center of the tube and mass flow averaged value compared with the TCCOOL simulation with a expansion factor* < 20 *of the mesh*

4.3 One tube with inlet manifold

The results for the tube has been presented in the previous chapter 4.2.2 and only the influence of placing an inlet manifold will be presented. Since the slightly lower heat flux, $q_j = 5.1495[MW/m^2]$, is used in this simulation the temperature and pressure will be slightly lower than for the *validation case 2*. The solution converged with a time scale of $5.0 \cdot 10^{-5}[s]$ after 12991 iterations. As can be seen in figure 4.3.1 the solution is more unstable than the previous simulations, the RMS value is experiencing a higher amplitude than the previous simulations. Since the solution is more unstable also the imbalance of the energy equation is unstable.

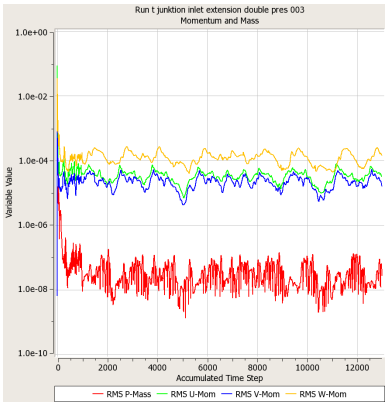
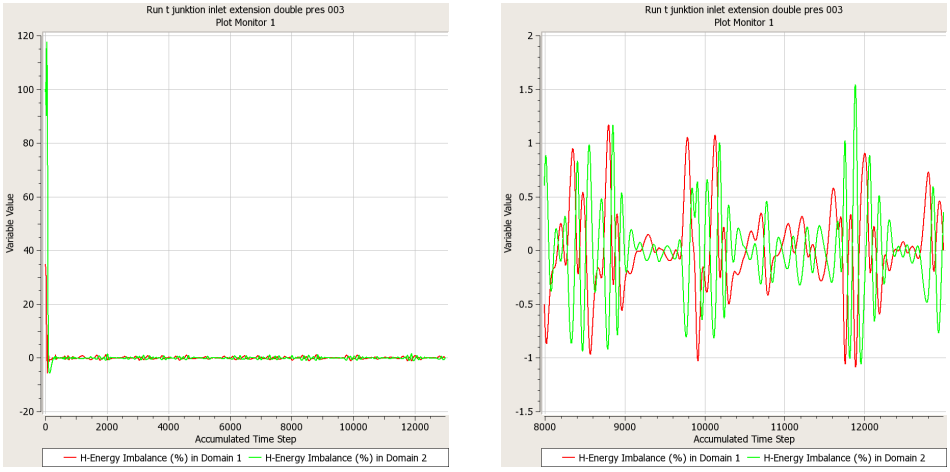


Figure 4.3.1 – *RMS residuals for mass - and momentum equation*

This behavior was seen in all the following simulations and it also increase in frequency, but decrease in amplitude. When the pressure decreases in the corner, as in figure 4.3.4, the flow will accelerate at that corner.



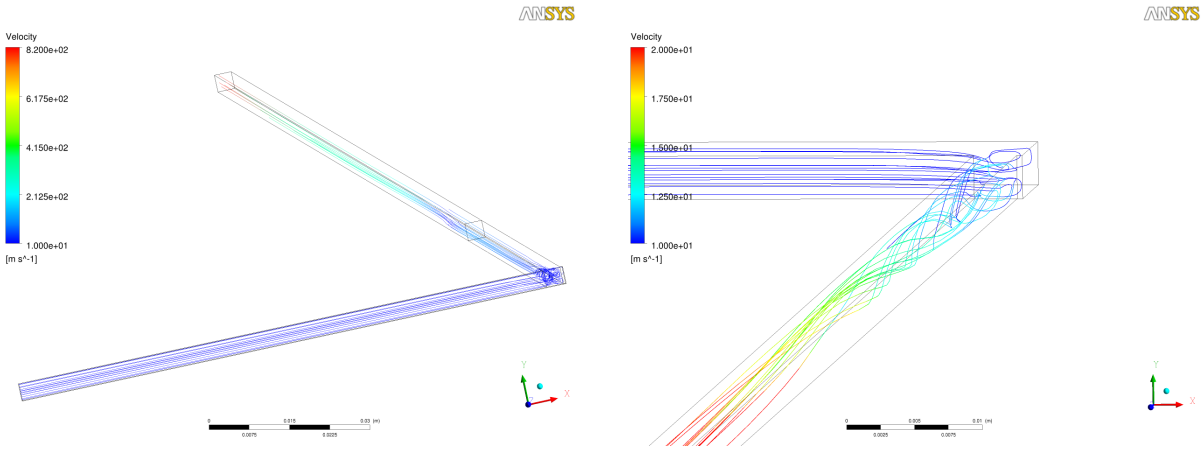
(a) *Full simulation* **(b)** *The 2000 most recent iterations*

Figure 4.3.2 – *Imbalance plot of the solution for energy equation*

When the flow is three dimensional the pressure gradient makes the flow swirl, as can be seen in figure 4.3.3.

Table 4.3.1 – Imbalance of the final values for the solution of simulation, chapter 3.4.1.

Imbalance			
Source		Fine region	Coarse region
Energy		0.0079%	0.3498%
Momentum	U	0.0%	0.0%
	V	0.0%	0.0%
	W	0.0%	0.0%
Mass		−0.2113%	0.5093%



(a) *Full tube with inlet manifold* **(b)** *Zoom of the corner of tube*

Figure 4.3.3 – *Streamlines coloured by velocity*

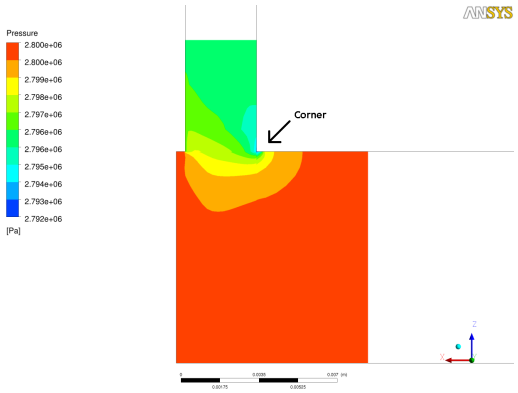
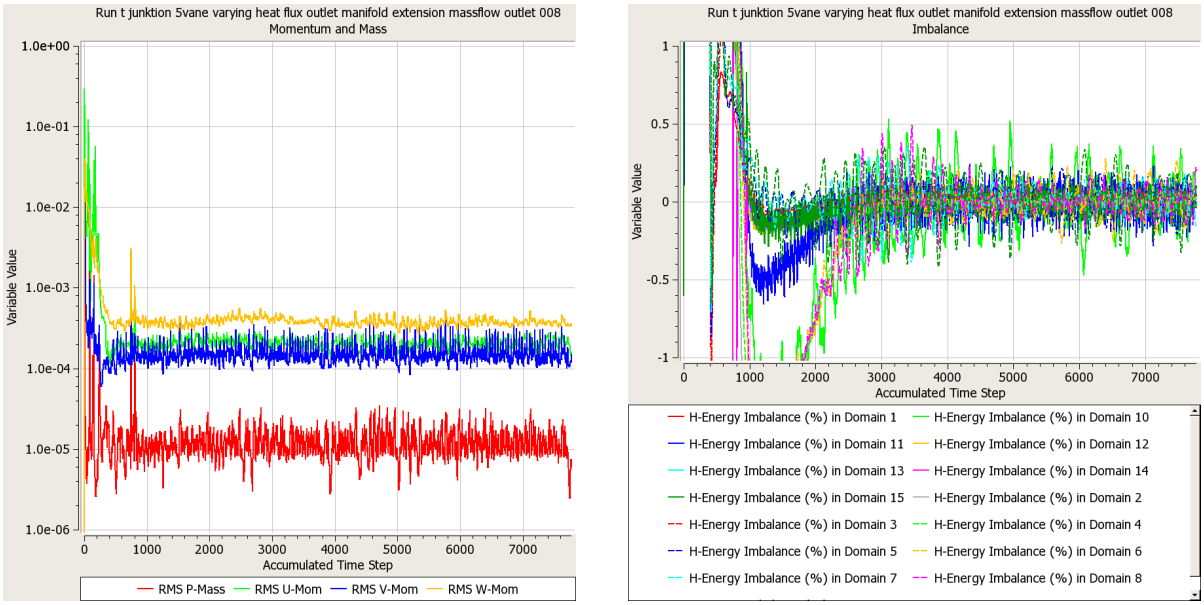


Figure 4.3.4 – *Contour plot of pressure in XZ-plane at center of channel, y=3.2[mm]*

4.4 Five tubes with inlet - and outlet manifold

The solution were considered converged after 7777 iterations with a timestep of $5 \cdot 10^{-5}[s]$. The RMS residuals had leveled out and the imbalance were sufficiently low, $< 0.5\%$, as can be seen in figures 4.4.1. In total the energy imbalance in the domains varied between $-0.1554\% - 0.18\%$. Unfortunately does not *CFX* give the total imbalance between the inlet and outlet, which would be interesting to see. As could be seen in the simulation



(a) **(b)**

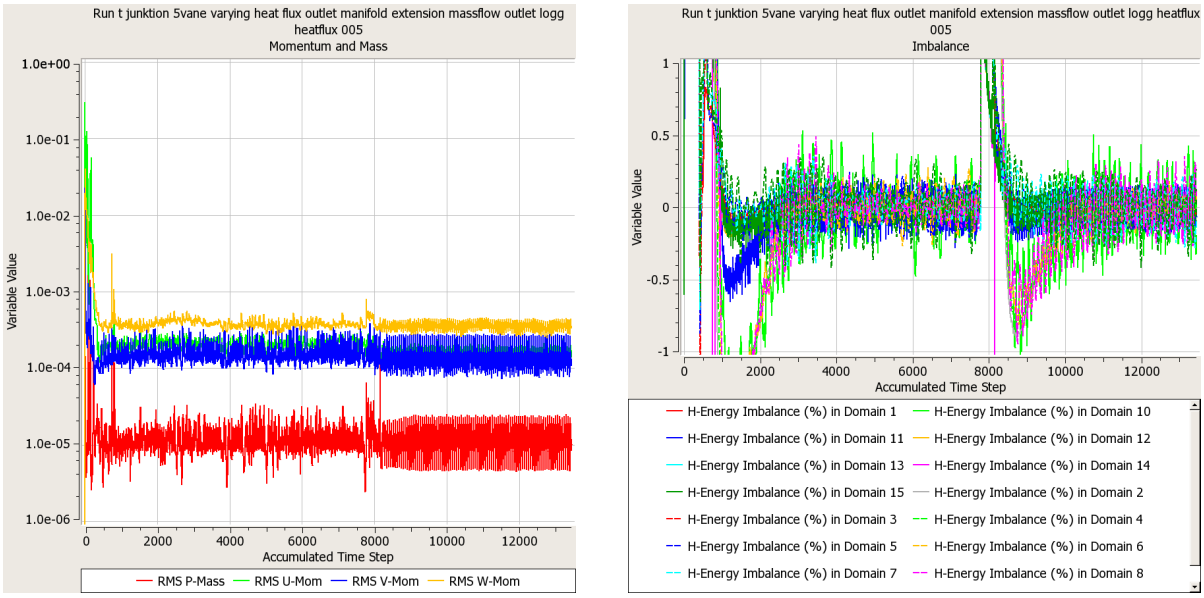
Figure 4.4.1 – *Solver: a) RMS residuals for mass- and momentum equation and b) Imbalance plot for the energy equation*

with one tube and only inlet manifold the flow is very unstable due to the amount of secondary flow. The simulation with inlet manifold and outlet manifold and 5 tubes took 67.6[h] of total CPU time with 8 cores in the Volvo Aero cluster.

The heat flux was changed to be varying after the solution had converged, i.e after 7777 iterations, the heat flux was set to vary with the mass flow. For this solution the residuals and imbalance are monitored together with the mass flow variation in each tube and the heat flux variation in each tube. The solution were considered converged after 5667 iterations and it took in total CPU time 47.9[h], with 8 cores in the Volvo Aero cluster, see figure 4.4.2.

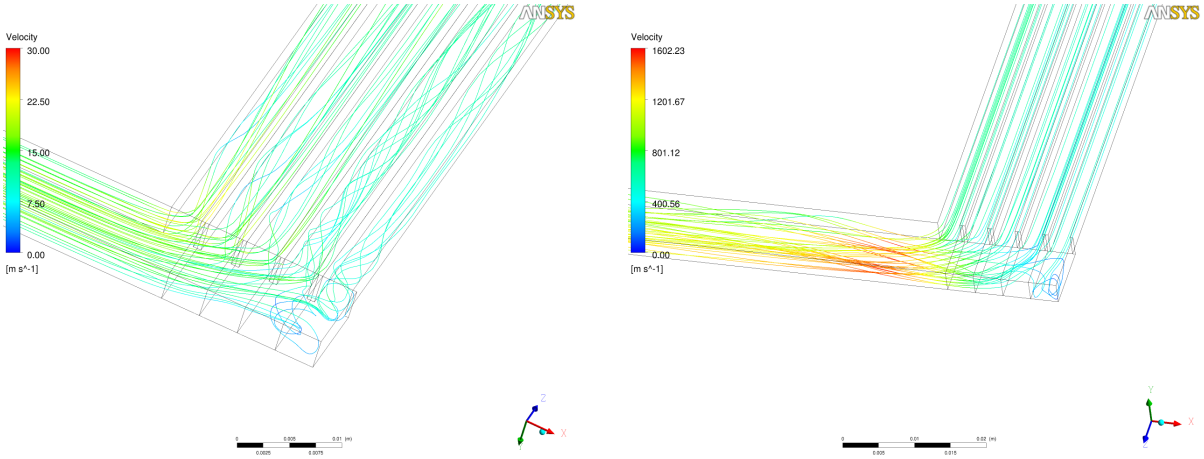
The general flow behavior can be seen in figure 4.4.3. As explained in the beginning of the section some secondary flow can be seen where the flow go down in the tubes. As can be seen in figure 4.4.4 the heat flux vary as defined with the boundary condition, $q_j(\dot{m})$, which creates a temperature variation between the tubes, seen in figure 4.4.5. This result shows that it is possible to have a variable cooling effect of the tubes.

In the previous results the velocity was most affected by the interface. But as seen in figure 4.4.6 when an interface is perfectly matched, meshes meeting are exactly the same, the velocity is not affected.



(a) **(b)**

Figure 4.4.2 – *Solver: a) RMS residuals for mass- and momentum equation and b) Imbalance plot for the energy equation*



(a) *Inlet region* **(b)** *Outlet region*

Figure 4.4.3 – *Streamlines coloured with velocity*

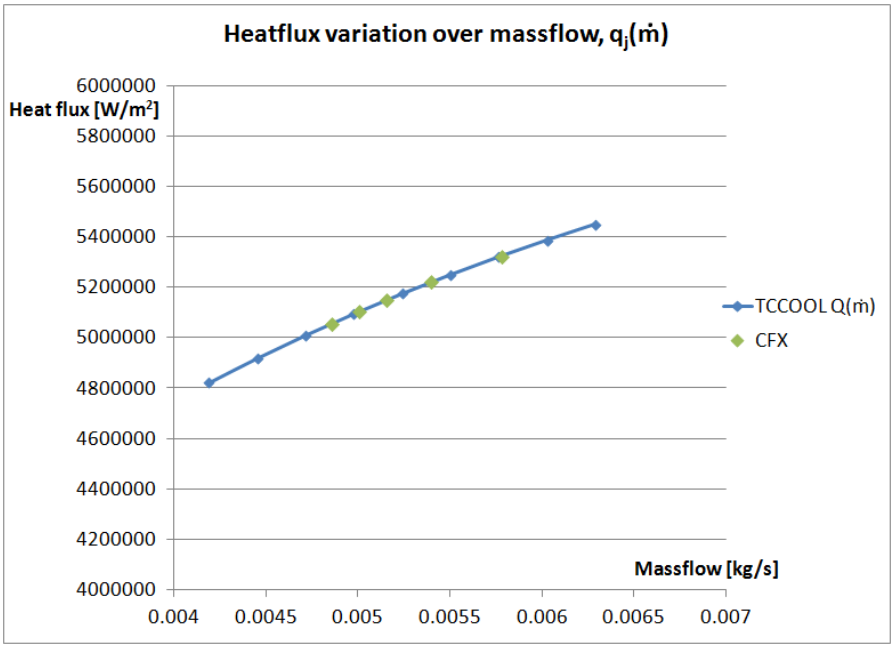


Figure 4.4.4 – *Plot comparing the TCCOOL simulation variation in heat flux depending on mass flow compared with results from CFX simulation*

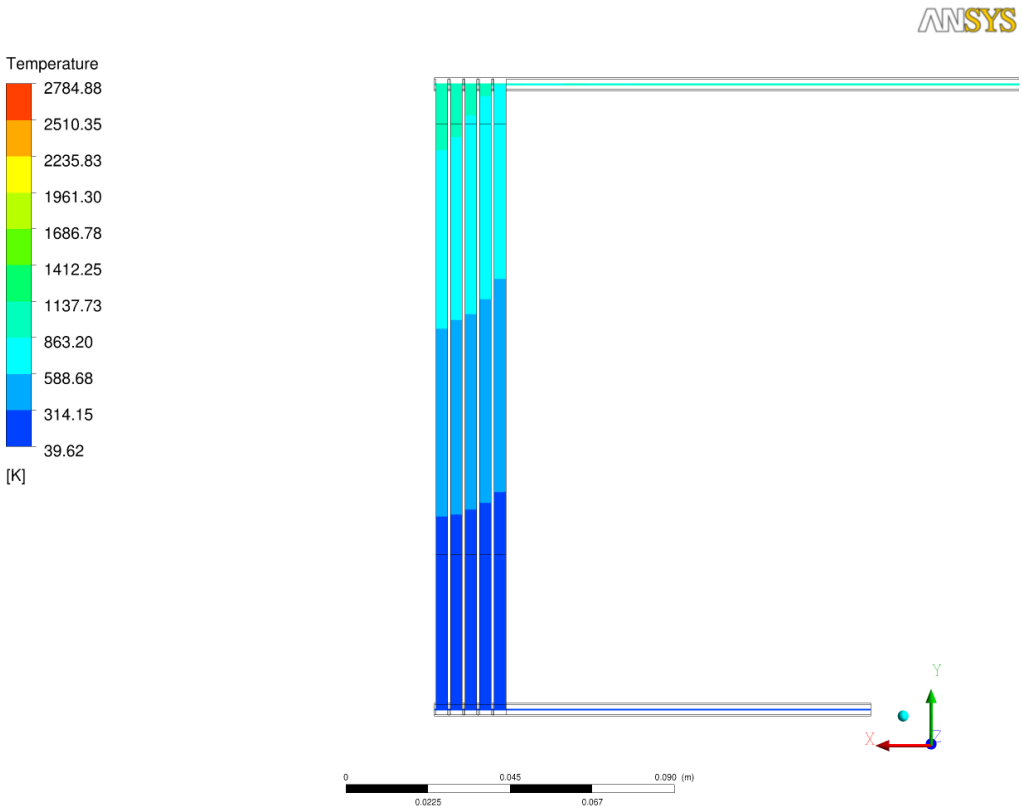
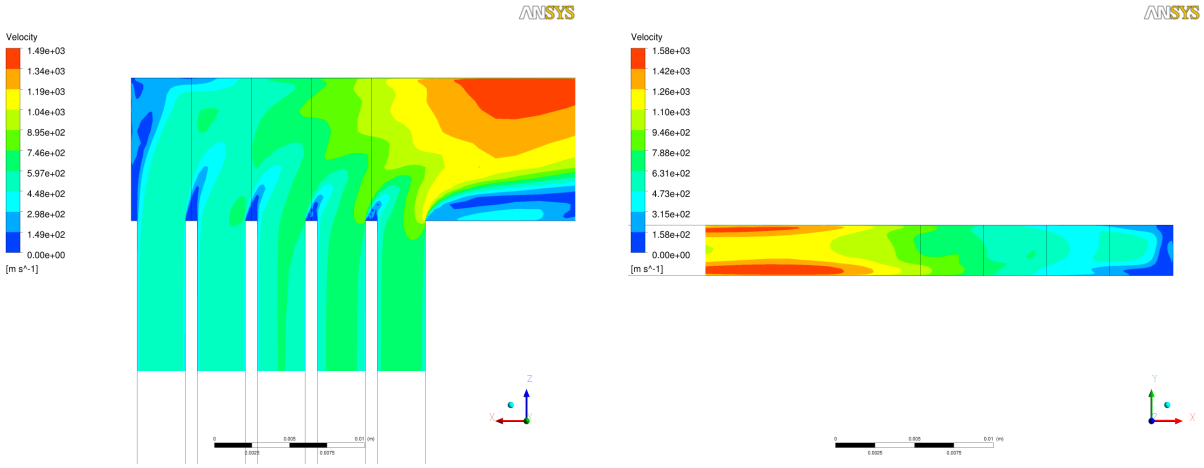


Figure 4.4.5 – *Contour plot of temperature in center of y-direction with XZ-plane, inlet is located at the foreground*



(a) *XZ-plane at center in y-direction* **(b)** *XY-plane at center in z-direction*

Figure 4.4.6 – *Contour plot of velocity in center of the outlet manifold*

4.4.1 Mass flow variations

From the figure 4.4.7 it is possible to see how much mass flow is going into each tube. The variation is showed in percent since the result from the simulation described in chapter 3.4.2, with a lower mass flow, also are included. It is seen that when a outlet manifold is present the difference in mass flow between each tube is much greater than when only letting the fluid flow out of the tube.

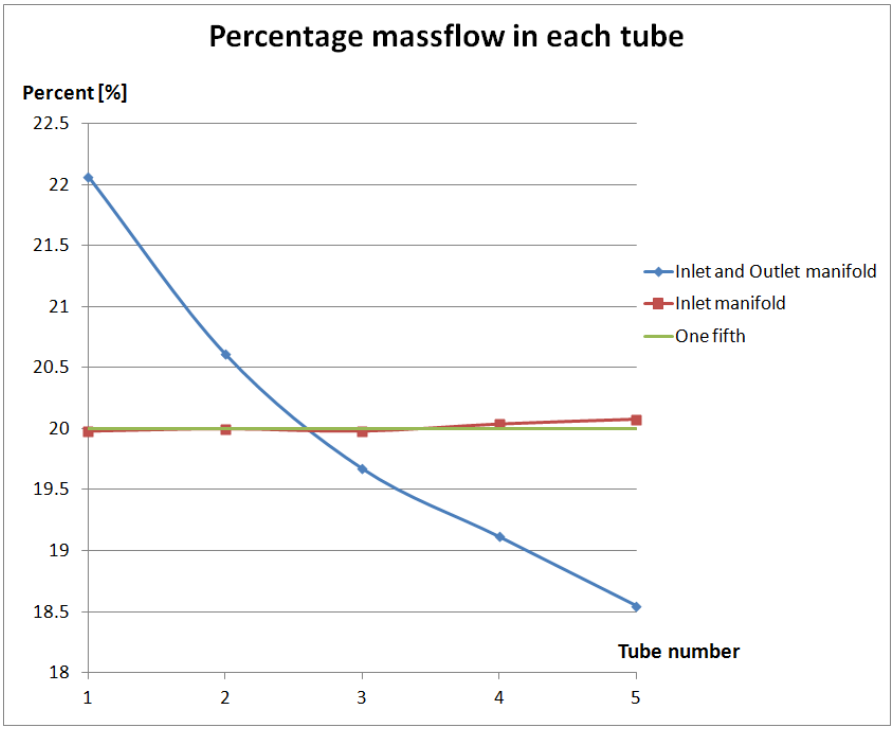


Figure 4.4.7 – *Mass flow variation in the tubes compared to the total mass flow, tubes are numbered from the tube closest to the inlet as tube number 1 and in following order to tube number 5.*

5 Conclusion

The results from this work show that it is possible to have a coarse mesh where the flow is not of interest and the only importance is the temperature increase and the pressure drop.

If the fine part of the tubes, explained in chapter 3.3.1, would have been used for the part of the tube where the coarse mesh is placed, the number of nodes would be $102400 \cdot 3.2[m] = 327680$ but the results show that with this coarse mesh it is possible to have a lot less, 240 nodes, which is the number of nodes used in the simulation described in chapter 3.3.1.

In the initial simulations a constant heat flux were used. It was a bit unclear whether the *.tload*-file, see figure 3.2.4, or the *Total heat flux* in the *TCCOOL* simulation should have been used in the beginning of the project. Later it was clear that the *Total heat flux* should have been used but only because of the inaccuracy in the approximation of the total heat flux from the *.tload*-file. Therefore it is uncertain how the results for the slightly lower heat flux will differ from the simulations with the slightly higher heat flux.

The level of turbulence and the scales for turbulence are not verified since the validation case is based on a one dimensional solver that does not include turbulent scales, it is therefore unknown how well the turbulent scales are transferred in this very coarse mesh.

As could be seen in the simulation with five tubes the solution takes very long time, and also creating CEL expressions that varies in the solution and that is depending on another variable which also is varying in the solution makes the solver converge slower, "varies in the solution" meaning that it varies in each cell but also in each iteration. Also a very low time step is needed to get a solution at all. The reason for the long simulation time may be because the boundary layer is fully resolved but also due to unsteadiness in the flow.

The solution for five tubes are unsteady because the solution is possibly a transient problem. The sharp edge creates a vortex that is unsteady as can be seen in figure 4.4.3. So the flow is not able to reach a steady solution and the flow is possibly flipping between the tubes, which also Volvo Aero has seen in other simulations.

The interfaces are a source of problem. While the mesh expansion factor is not an issue in the validation cases it creates a large discrepancy when the coarse mesh is connected to a fine mesh with a interface. But by smoothening the mesh out, as seen in chapter 4.2.4 the discrepancies are expected to completely disappear.

6 Recommendations and Future work

The validation software *TCCOOL* does not resolve the turbulent scales, since it is a one dimensional software. However, in *CFX*, the turbulent scales are modeled and for fully resolved flow it is known how they should behave. But for this very coarse mesh and subdomain it is not known how *CFX* treats the turbulent scales. Therefore it is necessary to see how well the turbulent scales are transferred in this very coarse mesh and how the simulation is affected.

Investigate what variables are affected by this very coarse mesh by testing different mesh resolutions. When the coarse mesh were produced the boundary conditions where a bit different compared to what was presented in this thesis, and finally used, so it would be good to investigate some more coarser mesh. For example removing a cell in the *x-direction* resulting in a resolution of *2x2* cells in x - y plane instead of the current *3x2* cells in x - y plane.

When the five tube simulation were run it showed some unstable behavior in the solution stage. This behavior was explained to be due to the solution actually was transient. Therefore it would be interesting to investigate how a transient solution would look like and how much this transient solution would differ from the steady-state solution produced in this work.

When the fine mesh were created it was just based on good judgment and were never tested for mesh independence. It would therefore be important to test this mesh for grid independence to be sure that the heat transfer and velocity is not affected by the mesh resolution.

Also since the first node in the fine mesh were mostly below $y^+ < 1$ the wall functions were never used. In order to further reduce computation time it is necessary to reduce the mesh resolution for the fine mesh so the wall functions are used in the fine mesh regions as well as in the coarse.

References

[1] J. D. Anderson. *Modern compressible flow*. 3rd ed. McGrawHill, 2004. ISBN: 007-124136-1.

[2] ANSYS. *ANSYS CFX-Solver Modeling Guide*. Version 14.0. ANSYS Inc., 2011.

[3] ANSYS. *ANSYS CFX-Solver Theory Guide*. Version 14.0. ANSYS Inc., 2011.

[4] D. A. Ashworth et al. "Unsteady aerodynamic and heat transfer processes in a transonic turbine stage". In: *Journal of Engineering for Gas Turbines and Power* 107 (1985), pp. 1022–1030.

[5] L. Bronx et al. *Hot testing of laser welded channel wall nozzles on vulcain 2 engine and subscale staged combustion demo*. Paper, IAC-11.C4.3.7. 5 October 2011. Volvo Aero Corporation.

[6] M. J. Cook. *An Evaluation of Computational Fluid Dynamics for Modelling Buoyancy-Driven Displacement Ventilation*. Thesis. Leicester, England: De Montfort University.

[7] L. Davidsson. *Fluid mechanics, turbulent flow and turbulence modeling*. Lecture notes. Göteborg, Sweden: Applied Mechanics. URL: `http://www.tfd.chalmers.se/Ëœlada/MoF/lecturenotes.html`.

[8] D. D. Ebbing and S. D. Gammon. *General Chemistry*. Charles Hartford, 2010. ISBN: 978-0-538-49752-7.

[9] A. Favre et al. *La turbulence en mécanique des fluides [Turbulence in fluid mechanics]*. .Équations fondamentales des fluids à masse volumique variable en écoulements turbulents [Fundamental equations of fluids with variable density in turbulent flow]. Gauthier-Villars, Paris, 1976.

[11] P. G. Hill and C. R. Peterson. *Mechanics and thermodynamics of propulsion*. 2nd ed. Addison-Wesley Publishing Company, Inc., 1992. ISBN: 0-201-14659-2.

[12] F. P. Incropera. *Fundamentals of heat and mass transfer*. 6th ed. Wiley, 2007. ISBN: 9780471457282.

[14] J. Larsson. *Numerical simulation of turbulent flows for turbine blade heat transfer applications*. PhD Thesis. Chalmers University of Technology: Thermo and fluid dynamics, 1998.

[15] E. W. Lemmon, M. L. Huber, and M. O. McLinden. *NIST Reference Fluid Thermodynamic and Transport Properties— REFPROP*. Version 8.0. Physical, Chemical Properties Division, National Institute of Standards, and Technology, **U.S. Department of Commerce**, 2008.

[17] F. R. Menter. "Influence of freestream values on $k-\omega$ turbulence model predictions". In: *AIAA JOURNAL* 30.6 (1992).

[18] F. R. Menter. "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications". In: *AIAA JOURNAL* 32.8 (1994).

[19] M. H. N. Naraghi. *User manual for RTE2002 Version 1. A computer code for three-dimensional rocket thermal evaluation*. Version 1. Tara Technologies, LLC, 2002.

[20] P. J. Oliveira. "A traceless stress tensor formultion for viscoelastic fluid flow". In: *Journal of Non-Newtonian Fluid Mechanics* 95 (2000), pp. 55–65.

[22] S. Sarkar and L. Balakrishnan. *Application of a reynold stress turbulence model to he compressible shear layer*. report. NASA Langley Research center, USA: Computer applications in science and engineering.

[23] H. Schlichting and K. Gersten. *Boundary-Layer Theory*. Springer, 1951.

[24] T. Serghides. "Estimate friction factor accurately". In: *Chemical engineering* 91 (5 1984), pp. 63–64.

[25] V. L. Streeter. *Handbook of Fluid Dynamics*. McGraw-Hill Book Company, inc., 1961.

[26] H. K. Veersteeg and W. Malalasekera. *An introduction to Computational Fluid Dynamics. the finite volume method*. Pearson Education Limited, 2007.

[27] F. M. White. *Fluid Mechanics*. McGraw Hill, 2008.

[28] D. C. Wilcox. "Reassessment of the Scale-Determining Equation for Advanced Turbulence Models". In: *AIAA JOURNAL* 26.11 (1988).

List of Figures

List of Tables

A Softwares

The full names of the different softwares used in this work are printed below:

Creating the geometry and mesh

ANSYS® ICEM CFD™ 14.0
Linux 64-bit
build date: October 12.2011
Version: 14.0
Copyright 2011, SAS IP, Inc.

Setting up the cases, applying boundary conditions, etc.

ANSYS ® CFX® -Pre
Version 14.0
© 2011 ANSYS, Inc.
ID: CFX-14.0 build 2011.10.10-22.59

Running the solution stage

ANSYS® CFX® -Solver Manager
Version 14.0
© 2011 ANSYS, Inc.
ID: CFX-14.0 build 2011.10.10-22.59

Post processing the results

ANSYS® CFX® -Post
Version 14.0
© 2011 ANSYS, Inc.
ID: CFX-14.0 build 2011.10.10-22.59

B ICEM basic meshing

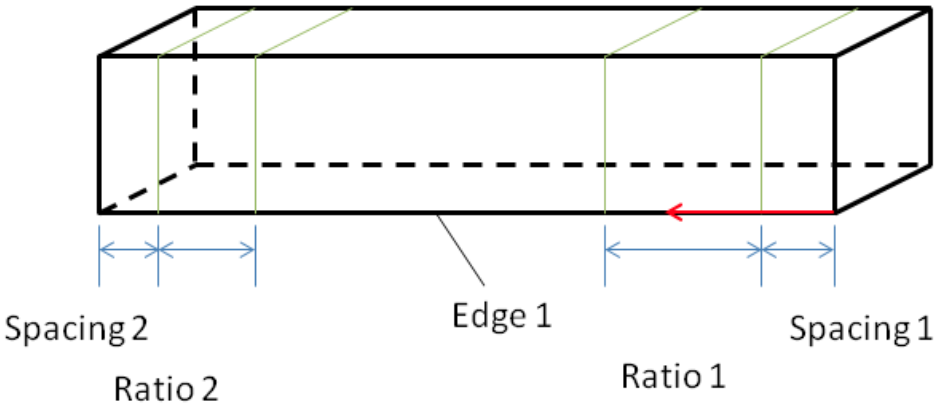


Figure B.0.1 – *ICEM meshing basics, green lines representing mesh, black lines representing geometry*

When meshing in *ICEM* the mesh is created out of blocks. The block has besides, a volume, vertices, faces and edges. For each edge the mesh parameters is set. With the Biexponential meshing law, used in this thesis the most, the spacing of the node intervals is calculated according to the law specified for Exponential 1 and 2. However, when Spacing 1, Ratio 1 and Spacing 2, Ratio 2 are used to define the distribution from the beginning of the edge to the middle of the edge, and Spacing 2 and Ratio 2 define the distribution from the terminating end of the edge to the midpoint of the edge. This bunching law is described by the following equation:

$$S_i = \int_0^i exp(a_0 + a_1 x + a_2 x^2 + a_3 x^3 + a_4 x^4) dx \tag{B.0.1}$$

From figure B.0.1 the beginning of the edge is the beginning of the red arrow and the direction points towards the end of the edge.

C Meshes for Mesh dependence

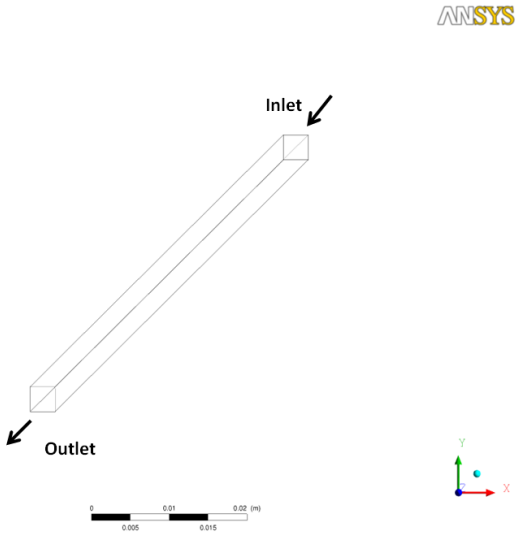


Figure C.0.1 – Mesh 1: *One single box, number of elements: 1, number of nodes: 8, Hexahedra*

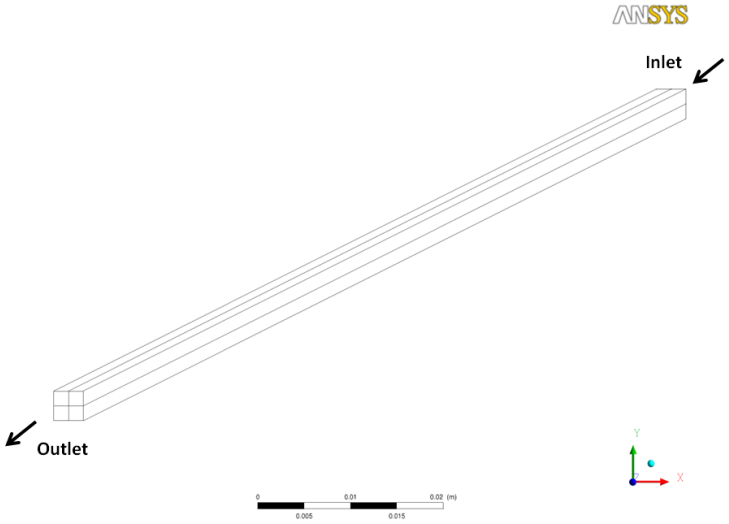


Figure C.0.2 – Mesh 2: *Number of elements: 4, number of nodes: 18, Hexahedra*

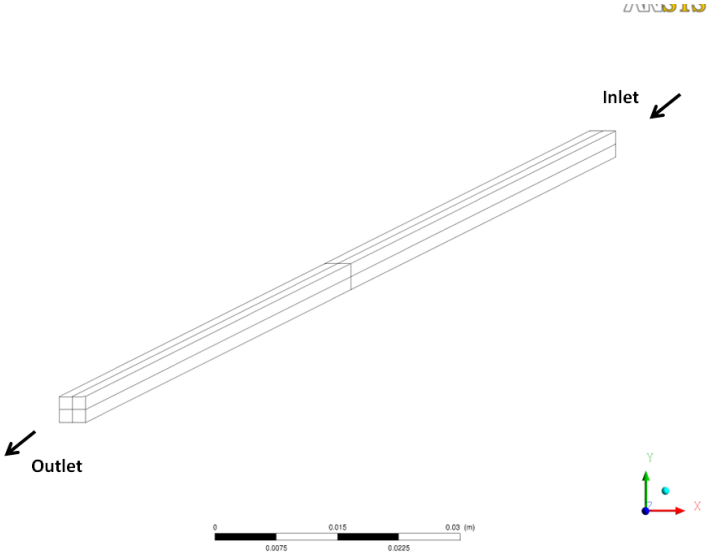


Figure C.0.3 – **Mesh 3:** *Number of elements: 8, number of nodes: 27, Hexahedra*

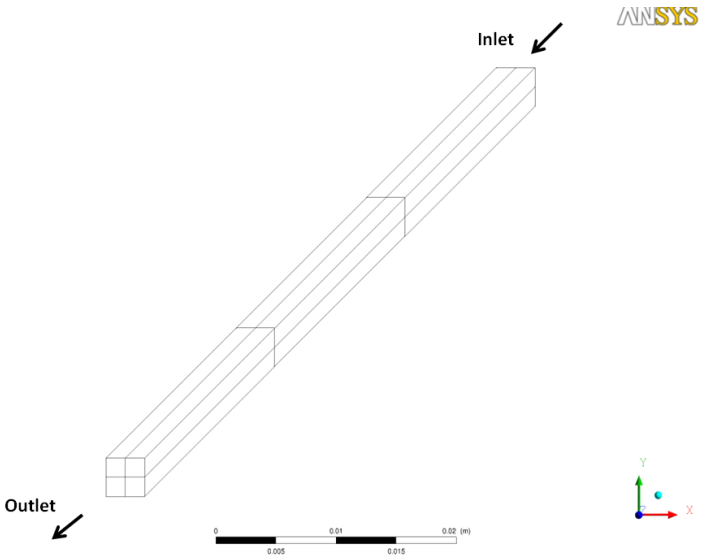


Figure C.0.4 – **Mesh 4:** *Number of elements: 12, number of nodes: 36, Hexahedra*

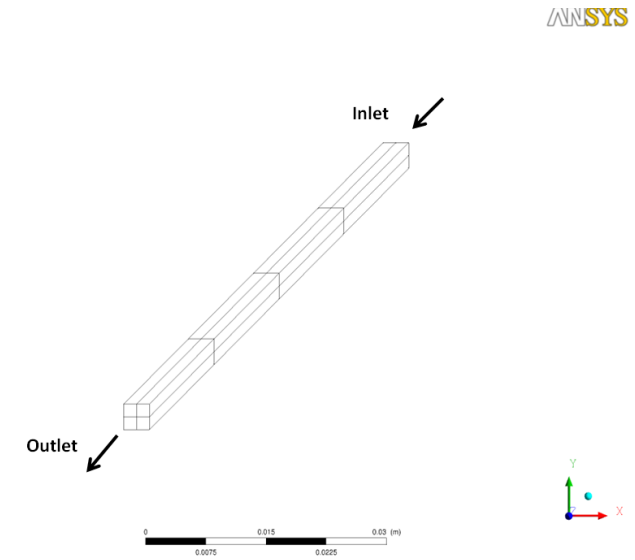


Figure C.0.5 – **Mesh 5:** *Number of elements: 16, number of nodes: 45, Hexahedra*

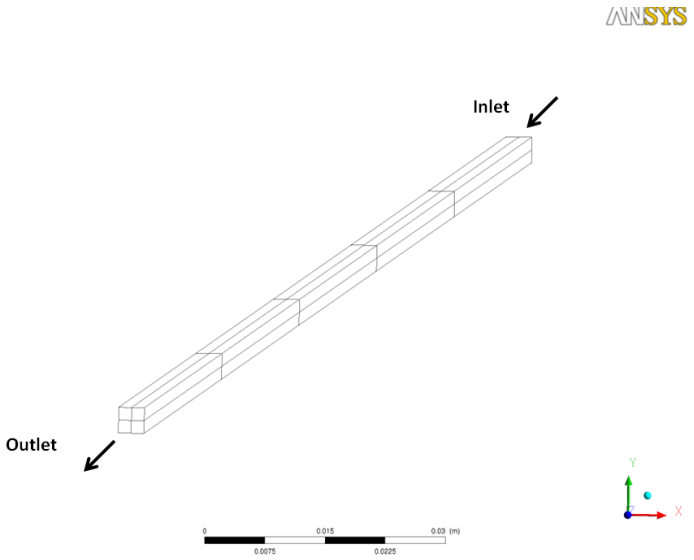


Figure C.0.6 – **Mesh 6:** *Number of elements: 20, number of nodes: 54, Hexahedra*

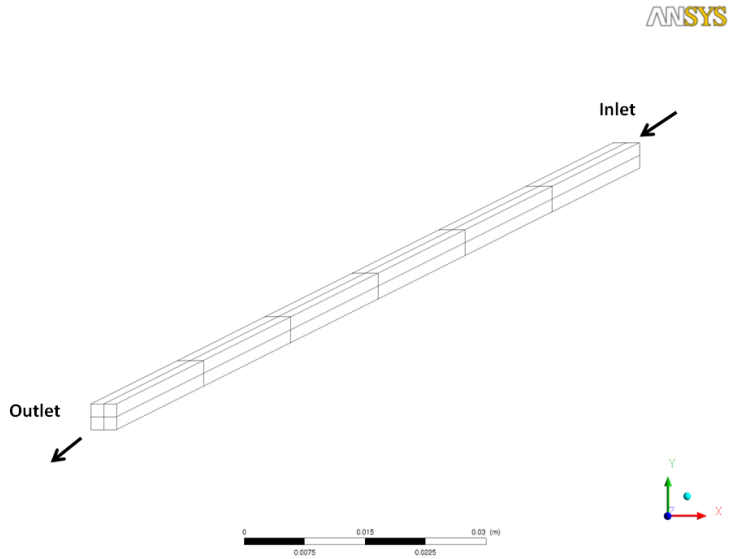


Figure C.0.7 – **Mesh 7:** *Number of elements: 24, number of nodes: 63, Hexahedra*