Numerical simulation of condensation in a supersonic nozzle for application to ejectors in refrigeration

by Marina Gallach Palma

This thesis is a CFD study of a supersonic nozzle for ejector refrigeration applications. Condensation of the working fluid is aimed inside the nozzle in order to study its consequences. As a first approach to verify the built-in Wet Steam model in ANSYS Fluent, a Laval nozzle is tested with water and compared with experimental data from Moses and Stein [1].

Thanks to the User Defined Wet Steam Property Functions in the User Defined Wet Steam model, the code is adapted for R134a both ideal and real gas properties and the study is done with perfect and real gas properties. Anyway, the software presents certain limitations and the real gas Peng-Robinson Equation Of State could not be implemented due to convergence problems.
Acknowledgements

This thesis would not have been possible without the help and advice of Yann Bartosiewicz. I would like to thank him for the patience and help guiding me along this project. Thanks to him I have learned a lot about this field which I might have never studied if it wasn’t for this thesis suggested by him.

To the friends that I have made in the department, who helped me going step by step.

To my friends who always supported me, and to Andreu for cheering on me to improve every day.

To my parents for their patience and support to go on.
# Contents

Abstract i

Acknowledgements iii

Contents v

List of Figures vii

List of Tables ix

Abbreviations xi

1 Introduction 1
   1.1 Context ................................. 1
   1.2 Ejector as a component ................. 2
   1.3 State of the art .......................... 5
   1.4 Objectives ............................... 7
      1.4.1 Thesis structure ..................... 7

2 Physical and Numerical Model 9
   2.1 Physical Model ........................... 9
   2.2 Wet Steam Model .......................... 11
      2.2.1 Flow equations ....................... 11
      2.2.2 Wet Steam Equation .................... 12
      2.2.3 Phase Change Model .................... 13
   2.3 RANS Turbulence Models ................. 15
      2.3.1 Standard k-ε Model .................... 16
      2.3.2 Shear-Stress Transport k-ω Model .... 17
   2.4 Wall approach ............................ 17
      2.4.1 Wall Function ......................... 18
      2.4.2 Wall Resolved .......................... 19
   2.5 Solver/Numerical setup ................... 19

3 Validation: Nozzle + Steam 21
   3.1 Description of the cases ............... 21
   3.2 Numerical Setup .......................... 22
      3.2.1 Mesh and Geometry ..................... 22
### Contents

3.2.2 Boundary Conditions .............................................. 22
3.3 Results I ................................................................. 23
  3.3.1 Experiments No. 410-421 ......................................... 23
  3.3.2 Turbulence .......................................................... 28
3.4 Results II ................................................................. 29
  3.4.1 $\Delta T_{\text{saturation}}$ variations ............................... 29
  3.4.2 Shock Waves ....................................................... 31

4 Validation: Nozzle + Refrigerant ................................. 35
  4.1 Description of the cases ............................................ 35
    4.1.1 Refrigerant ..................................................... 35
  4.2 Numerical Setup .................................................... 36
    4.2.1 Geometry and Mesh ........................................... 36
    4.2.2 Boundary Conditions ........................................ 36
    4.2.3 R134a User Defined Functions .............................. 37
      4.2.3.1 Ideal Gas Properties ................................ 38
      4.2.3.2 Real Gas Properties .................................. 40
  4.3 Results ................................................................. 42
    4.3.1 Fully supersonic .............................................. 43
    4.3.2 Shock waves ................................................... 48

5 Conclusions ............................................................. 51
  5.1 Conclusions ........................................................... 51
  5.2 Futur Work ............................................................ 52

A Property functions for R134a ....................................... 53
  A.1 Saturation pressure function ................................... 53
  A.2 Vapor property functions for real gas ....................... 53
    A.2.1 $C_p-v$ ....................................................... 53
    A.2.2 $C_v-v$ ....................................................... 53
    A.2.3 $\rho_v$ ....................................................... 53
    A.2.4 $s_v$ .......................................................... 54
    A.2.5 $mu_v$ .......................................................... 54
    A.2.6 $k_{tv}$ .......................................................... 54

B User Defined Functions for R134a ................................ 55
  B.1 UDF for ideal gas ................................................... 55
    B.1.1 Ideal Gas EOS - Ideal Gas Properties .................... 55
    B.1.2 Ideal Gas EOS - Real Gas Properties ..................... 62
  B.2 UDF for real gas .................................................... 70

Bibliography .............................................................. 79
# List of Figures

1.1 Example of an ejector in a solar air-conditioning cycle .......................... 2  
1.2 Qualitative P and v of an ejector ....................................................... 3  
1.3 Ejector Characteristic Curve .............................................................. 4  
1.4 Temperature entropy diagram [2] ......................................................... 4  
  
2.1 Nozzle operation regimes depending on inlet pressure from [3] ............... 10  
2.2 Temperature - Entropy Plots .............................................................. 11  
2.3 Divisions of the near-wall region ....................................................... 18  
  
3.1 The 2D Laval nozzle .......................................................... 22  
3.2 Pressure ratio - Exp. 410 and 421 ..................................................... 24  
3.3 Liquid Mass Fraction - Exp. 410 and 421 ........................................... 24  
3.4 $\Delta T_{sat}$ - Exp. 410 and 421 ......................................................... 25  
3.5 Logarithm Droplet Nucleation Rate - Exp. 410 and 421 ............................ 26  
3.6 Pressure ratio of inviscid and experimental results for Exp No. 410 ...... 26  
3.7 T-s plots with Wet, Dry and Perfect gas cases .................................... 27  
3.8 Contour Static Pressure [Pa] - Exp. 410 ................................................. 27  
3.9 Contour Logarithm Droplet Nucleation Rate [-] - Exp. 410 ..................... 28  
3.10 Pressure ratio plot comparing turbulence models - WS .......................... 28  
3.11 Liquid mass fraction plot comparing turbulence models - WS ................ 29  
3.12 Pressure ratio of inviscid cases - $\Delta T_{sat}$ ...................................... 30  
3.13 Liquid Mass Fraction of inviscid cases - $\Delta T_{sat}$ .............................. 30  
3.14 $\Delta T_{sat}$ of inviscid cases - $\Delta T_{sat}$ ............................................. 31  
3.15 Pressure Ratio and $\Delta T_{sat}$ - SW .................................................. 32  
3.16 LMF and Log10(DNR) - SW .............................................................. 32  
3.17 Contour Static Pressure [Pa] Wet k-$\varepsilon$ - SW ................................. 33  
3.18 Log10(Droplet nucleation rate) contour plot, Wet k-$\varepsilon$ - SW ............ 33  
3.19 Liquid Mass Fraction [-] contour plot Wet k-$\varepsilon$ - SW ....................... 34  
  
4.1 Pressure density plot at constant temperature ...................................... 40  
4.2 Qualitative P - v Plot for cubic EOS from [4] ...................................... 41  
4.3 Mach number plots for FS cases - PG .................................................. 44  
4.4 Pressure ratio plots for FS cases - PG ................................................ 44  
4.5 Liquid Mass Fraction plots for FS cases - PG ....................................... 45  
4.6 Log10(Droplet Nucleation Ratio) for Fully Supersonic cases -PG ........... 45  
4.7 $\Delta T_{sat}$ for FS cases - PG ............................................................. 46  
4.8 $\Delta T_{sat}$ for FS cases - PG & HG ...................................................... 47
List of Figures

4.9  Contour plot for Liquid Mass Fraction [-] for $\Delta T_{sat} = -3$ K Fully Supersonic cases - HG .......................................................... 47
4.10 Pressure ratio for SW cases - R134a ................................................. 49
4.11 Mach number for SW cases - R134a .................................................. 49
4.12 Contour plot for Liquid Mass Fraction [-] for $\Delta T_{sat} = -3$ K SW cases - HG 50
4.13 $\Delta T_{sat}$ plot for SW cases .............................................................. 50
List of Tables

1.1 Vapor quality for wetness evaluation ............................. 5
3.1 Geometry measurement of the mesh (half of the real nozzle) .......... 22
3.2 Boundary conditions given by [1] and [5] .......................... 23
3.3 Boundary conditions for shock waves in Wet Steam .................. 23
3.4 Mass flow rate values in Exp. No. 410-421 cases - WS ............... 27
3.5 Mass flow rate values in Turbulence cases - WS ................... 29
3.6 Mass flow rate values in $\Delta T_{\text{sat}}$ cases - WS ................. 31
3.7 Mass flow rate values in Shock Wave cases - Wet Steam ............. 33
4.1 Boundary conditions for fully supersonic cases - R134a ............... 37
4.2 Boundary conditions for shock wave cases - R134a .................. 37
4.3 Mass flow rate values in Fully supersonic Perfect and Hybrid cases - R134a 48
4.4 Mass flow rate values in SW PG and HG .......................... 49
## Abbreviations

<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFD</td>
<td>Computational Fluid Dynamics</td>
</tr>
<tr>
<td>DS</td>
<td>Dry Steam</td>
</tr>
<tr>
<td>EOS</td>
<td>Equation Of State</td>
</tr>
<tr>
<td>EWT</td>
<td>Enhanced Wall Treatment</td>
</tr>
<tr>
<td>FS</td>
<td>Fully Supersonic</td>
</tr>
<tr>
<td>GWP</td>
<td>Global Warming Potential</td>
</tr>
<tr>
<td>HG</td>
<td>Hybrid Gas</td>
</tr>
<tr>
<td>ODP</td>
<td>Ozone Depletion Potential</td>
</tr>
<tr>
<td>P</td>
<td>Pressure</td>
</tr>
<tr>
<td>PG</td>
<td>Perfect Gas</td>
</tr>
<tr>
<td>RANS</td>
<td>Reynolds Averaged Navier-Stokes</td>
</tr>
<tr>
<td>SST</td>
<td>Shear Stress Transport</td>
</tr>
<tr>
<td>SW</td>
<td>Shock Waves</td>
</tr>
<tr>
<td>T</td>
<td>Temperature</td>
</tr>
<tr>
<td>UDF</td>
<td>User Defined Functions</td>
</tr>
<tr>
<td>UDWSF</td>
<td>User Defined Wet Steam Property Functions</td>
</tr>
<tr>
<td>WS</td>
<td>Wet Steam</td>
</tr>
</tbody>
</table>
Chapter 1

Introduction

1.1 Context

Refrigeration technologies are significant for food processing and also for human thermal comfort, among others. Due to global warming, the world’s warmest regions are expected to increase electricity consumption for cooling systems. As a result, more fossil fuels will be consumed leading to environmental pollution.

New cooling technologies, such as ejector refrigeration systems (ERS) [6], are being developed to avoid the use of primary energy. ERS provide mechanical simplicity together with less electricity consumption in comparison with traditional vapor compression refrigeration systems. In addition, these newer systems consider the usage of low-quality heat to activate the cycle and produce cooling. Many industries such as food or chemical processing drop huge loads of heat to the environment when otherwise these loads could be used for refrigeration that they might need after all.

Ejector-based heat driven circuits offer off-grid properties; for example a solar thermal panel could be used as a heat source for the boiler and a photovoltaic panel could supply the electricity to run the pump in the cycle, as seen in Figure 1.1. In these cycles, the ejector works as a compressor but without any moving parts thus having low life-cycle maintenance.

Nevertheless, ejectors provide low efficiency properties in these systems and this has made them less abundant in the market. The main parameter that displays the efficiency
of a cycle is the Coefficient Of Performance (COP):

\[
\text{COP} = \frac{\text{cooling effect at the evaporator}}{\text{energy input at the boiler and pump}} = \frac{Q_e}{Q_b + W_p} \quad (1.1)
\]

where \( Q_e \) and \( Q_b \) are the cooling capacity at the evaporator and energy input to the boiler, and \( W_p \) is the work consumed by the mechanical pump.

Another parameter which is relevant for the ejector performance is the entrainment ratio, \( \omega \), and it is defined as:

\[
\omega = \frac{\dot{m}_s}{\dot{m}_p} \quad (1.2)
\]

where \( \dot{m}_s \) and \( \dot{m}_p \) are the mass flow rates for secondary and primary flows respectively.

### 1.2 Ejector as a component

The ejector is the main component in the ERS. The purpose of the ejector is to mix and compress two fluid flows that enter through the two inlets, which are at different energy levels. The qualitative values for \( P \) and \( v \), while the operation of an ejector, are illustrated in Figure 1.2. The motive or primary flow goes inside the ejector (M) through the convergent-divergent nozzle expanding and reaching supersonic velocity after the throat (M→i→ii). Furthermore, the suction or secondary flow (S) is drawn
into the supersonic motive jet. In the mixing section (ii→iii→iv) there is recompression of the flows, producing one only flow at supersonic velocity. When reaching the diffuser (iv→v→vi), the mixed flow must adjust to the outlet conditions and a series of shocks and complex interactions take place (in Figure 1.2 there is one only shock as it is an idealized case).

For every ejector and fluid there is a characteristic curve which describes its operating conditions [2]. An example can be seen in Figure 1.3. When the flow in the mixing section of the ejector is supersonic, the ejector is in ”on-design” conditions, and the flow is said to be choked. In these conditions, the entrainment ratio is maximum, and the suction mass flow rate is independent of the back or outlet pressure. Otherwise, the ejector is ”off-design”, when the secondary flow remains subsonic and the suction mass flow rate does depend on the outlet pressure. As seen in Figure 1.3, if back pressure lightly increases there is a huge reduction of the secondary mass flow rate, which is not a desired effect.

A relevant issue for the ejector performance is whether the fluid that is going to be used is prone to condensation or not. Small liquid droplets during the expansion might interfere with pressure and temperature. Regarding this situation, fluids could be categorized as dry or wet. The main difference between them is showed in the slope of their T-s diagrams; for a wet fluid the saturated vapour line has a negative slope whereas for a dry fluid the slope is positive. In other words, for a dry fluid, in an expansion process
starting in a saturated vapor point the final point will be outside of the dome and there will not be any condensation. In Figure 1.4 the shape of four different refrigerants is seen in a T-s diagram. R134a, water and propane have negative slopes so they are called wet fluids. Contrarily, R245fa is a dry fluid, as it has a slightly positive slope for medium-high temperatures.
In order to evaluate the level of wetness of a fluid, a new quantitative theoretical variable is proposed here. The way to obtain the wetness factor of a fluid is the following:

- Starting from a pressure which is 80% of the critical pressure $P_c$
- Considering an isentropic ($\Delta s = 0$) expansion
- Hypothesizing a pressure ratio of 4.36 ($P_{\text{initial}}/P_{\text{final}}$)
- Finally calculating the quality of the vapor in that final state

The calculations to find the wetness factor has been made for four different fluids: water, R134a, R245fa and propane. The results are found in Table 1.1. This factor could be understood as the % of liquid at the end of the expansion. This shows that water is actually the most “wet” fluid, after comes the R134a and the propane. R245fa is a wet fluid and in this theoretical process it does not condense.

<table>
<thead>
<tr>
<th>Fluid</th>
<th>Vapor Quality</th>
<th>Liquid Quality</th>
<th>Liquid %</th>
</tr>
</thead>
<tbody>
<tr>
<td>R134a</td>
<td>0.9154</td>
<td>0.0846</td>
<td>8.46</td>
</tr>
<tr>
<td>R245fa</td>
<td>Superheated</td>
<td></td>
<td>0</td>
</tr>
<tr>
<td>Propane</td>
<td>0.9205</td>
<td>0.0795</td>
<td>7.95</td>
</tr>
<tr>
<td>Water</td>
<td>0.7141</td>
<td>0.2859</td>
<td>28.59</td>
</tr>
</tbody>
</table>

Table 1.1: Vapor quality for wetness evaluation

It should be taken into account that the starting state point of the expansion influences the level of wetness at the end of the expansion, so this values are just illustrative for the conditions above.

### 1.3 State of the art

Many studies such as Little et al. in [7] (2015), numerically and experimentally studied an air ejector and agree that global properties are well predicted by the different turbulence models, but these fail to explain flow local features. Bartosiewicz et al. in [8] (2005) conclude that local phenomena prediction like shock wave location are not well determined yet.

Al-Ansary and Jeter [9] (2004) included water droplets into the air flow to reduce the irreversibles in the mixing chamber of the ejector. Also Hemidi et al. in [10] and [11]
(2009) had confirmed the difference between local and global properties. However and in the same way as Al-Ansary and Jeter, these last two papers worked with air ejector adding fine water droplets with an atomizer in order to simulate a two-phase flow. They insisted that different physical properties such as speed of sound might change with the water in air, which could modify some conditions.

Moses and Stein in [1] (1978) experimentally and numerically studied the growth of water droplets in a Laval nozzle in order to predict condensation onset. They compared the quantity of condensed liquid which comes from nucleation rate or droplet growth laws. Later, Zori and Kelecy [5] (2005) study wet steam condensation in ANSYS Fluent also in a Laval nozzle comparing the results with experimental data from Moses and Stein.

Using refrigerant in simulations there is Elakhdar et al. [12] (2011) who performed a simulation of an ejector with a refrigerant in vapor phase, but with a one-dimensional mathematical model. Some studies such as the one made by Mazzelli and Milazzo [13] (2014) avoided condensation issues and used dry fluids such as R245fa, eluding the two-phase modelling. Bartosiewicz et al. [14] (2006) compared their CFD ejector modelling with available 1-D data with R142b, but without condensation. They stated that 1-D is not enough to predict operation at different conditions and suggested that an experimental-CFD integration was needed in order to verify global and local properties.

Regarding the performance in the ejector, the studies made by Al-Ansary and Jeter [9] and Hemidi et al. [10] hypothesized that the presence of liquid droplets at the end of the nozzle can improve the entrainment effect of the ejector. In both cases, they identified an improvement in ejector performance when the flow was not choked (in off-design conditions). In [10], this improvement is noted with the increase of about 10 to 40% of the entrainment ratio depending on the conditions. In any case, there was no enhancement in the on-design operating.

Little and Garimella [15] (2015) performed an experimental study with R134a in an ejector and demonstrated that there is actually an improvement in COP up to a 12% when the inlet conditions were more saturated. Experimental research with real refrigerant ejector was also performed by Zegenhagen and Ziegler [16] (2015). They used R134a and due the boundary conditions applied they avoid condensation effects inside the ejector.
Chapter 1. Introduction

1.4 Objectives

This thesis was suggested due to the scarcity of research on condensation simulation in ejectors. The aim of this thesis is to study the condensation effects on a nozzle with a real refrigerant such as R134a. This study is targeted to later be a tool to predict behaviour of a full ejector with real gas as working fluid.

The main objective is to study the behaviour with condensation inside the nozzle. To do so, the first step is to verify the built-in Wet Steam model in ANSYS Fluent simulating software, which has water as fluid. After, with this model and the R134a refrigerant properties inside the User Defined Wet Steam Property Functions (UDWSPF) the condensation consequences on the nozzle are going to be studied.

The final step of the thesis was thought to be simulating real gas properties in a full ejector developed in Georgia Tech, but due to programming issues that will be presented in the following sections, reaching that point was not possible.

1.4.1 Thesis structure

The thesis is divided in different chapters and each one is briefly described below.

In Chapter 2, the physical model inside the nozzle is described. The numerical model and general set up for the Wet Steam build-in model are also detailed.

Chapter 3 reports the description of the first part of this thesis. It includes the cases’ features, its numerical set up and results for the nozzle study using water as refrigerant liquid, considering ideal gas properties. This Chapter was thought to be the verification of the built-in Wet Steam model in Fluent.

Chapter 4 reports the simulations of the nozzle using R134a as a refrigerant, including the definition of the different cases, the numerical set up and the results. In this Chapter, by using the user defined functions (UDF) the properties of the refrigerant are introduced into the simulations.

In Chapter 5, the conclusions and future works to continue the research are outlined.
Chapter 2

Physical and Numerical Model

2.1 Physical Model

The usual operation of a nozzle is illustrated in Figure 2.1. The fluid flows from a high pressure inlet to a lower pressure outlet. The plot shows the pressure ratio (pressure over inlet pressure) depending on the position along the nozzle. The flow can have different patterns depending on the boundary conditions and these are described below according to Figure 2.1:

a **Subsonic flow:** The back pressure is too high to induce sonic flow in the throat. The flow in the nozzle is subsonic. The outlet pressure is equal to the discharge pressure and the exit jet is subsonic.

b **Flow just chocked:** The throat becomes sonic and the mass flux reaches its maximum. Downstream, the flow is subsonic including exit jet.

c **Shock in nozzle:** The discharge pressure lies between pressures from conditions (b) and (f), there is a normal-shock wave in the diverging section in order to cause a subsonic flow back to the discharge condition. The throat is chocked and the mass flow remains maximum.

d **Shock at exit:** The discharge pressure forces a normal-shock wave just at the outlet. The flow is supersonic from the throat until the outlet, where after the shock becomes subsonic.
e **Overexpanded**: The flow compresses outside the outlet with a series of complex oblique shocks to reach the outlet pressure. The flow is supersonic downstream the throat and until the shocks.

f **Design condition**: The diverging flow is completely supersonic including the exit jet flow. The pressure ratio is called design pressure ratio of the nozzle for supersonic conditions.

g **Underexpanded**: The flow in the outlet expands in a series of complex supersonic waves until it reaches the outlet pressure.

Figure 2.1: Nozzle operation regimes depending on inlet pressure from [3]

Due to the expansion at the converging-diverging nozzle, and if the inlet conditions are close enough to the dome, condensation might appear. It will only happen in case that the working fluid is actually a wet fluid, as described in Chapter 1.

Anyway, condensation might not appear at the exact moment that the state path is crossing the saturation dome since the fluid might be in a metastable or non-equilibrium state, states which are not represented in the equilibrium diagrams. This is because fluid particles need some relaxation time to get to the equilibrium state. This delay to reach condensation is seen in Figure 2.2; after a while having crossed the dome, liquid droplets begin to appear and the temperature increases due to the latent heat release.
2.2 Wet Steam Model

This section presents the tool that has been used along the thesis, ANSYS Fluent 14.5 Academic Version.

2.2.1 Flow equations

The flow is governed by the continuity, momentum, energy and transport compressible Navier-Stokes equations. These are coupled in a system of equations and it is solved by means of a density-based solver with control volume approach for the variables in $Q$. The equations are given in vector form by:

$$\frac{\partial W}{\partial Q} \frac{\partial}{\partial t} \int_V QdV + \oint [F - G]dA = \int_V HdV$$  \hspace{1cm} (2.1)

where the vectors $W$, source terms, $Q$, mixture quantities of primitive flow variables, $F$, advection terms, and $G$, diffusion terms, are defined as:

$$W = \begin{cases} 
\rho \\ 
\rho u \\ 
\rho v \\ 
\rho w \\ 
\rho E 
\end{cases}, \quad Q = \begin{cases} 
\rho \\ 
u \\ 
\rho u + p\hat{i} \\ 
\rho v + p\hat{j} \\ 
\rho w + p\hat{k} \\ 
\rho vE + \rho v
\end{cases}, \quad F = \begin{cases} 
0 \\ 
\tau_{xi} \\ 
\tau_{yi} \\ 
\tau_{zi} \\ 
\tau_{ij}v_j + q
\end{cases}$$  \hspace{1cm} (2.2)
The total energy, $E$, is defined as:

$$ E = h - \frac{p}{\rho} + \frac{v^2}{2} $$

(2.3)

where the Jacobian $\frac{\partial W}{\partial Q}$ is given by:

$$ \frac{\partial W}{\partial Q} = \begin{bmatrix} \rho_p & 0 & 0 & 0 & \rho_T \\ \rho_p u & \rho & 0 & 0 & \rho_T u \\ \rho_p v & 0 & \rho & 0 & \rho_T v \\ \rho_p w & 0 & 0 & \rho & \rho_T w \\ \rho_p H - \delta & \rho u & \rho v & \rho w & \rho_T H + \rho C_p \end{bmatrix} $$

(2.4)

where

$$ \rho_p = \left. \frac{\partial p}{\partial p} \right|_T, \rho_T = \left. \frac{\partial p}{\partial T} \right|_p $$

(2.5)

Two additional transport equations are introduced to solve the droplet formation and the mass transference between the two phases. Those are explained in the section 2.2.2.

This change in primary variables, described as preconditioning in [17], with respect to the usual Navier-Stokes equation ($\frac{\partial}{\partial t} \int_V WdV + \oint [F - G]dA = \int_V HdV$), is performed in order to overcome the numerical stiffness at low Mach number areas due to the gap between the fluid velocity $v$ and the speed of sound $c$, that is when the fluid is weakly compressible. In these zones, the stiffness of the equations leads to convergence problems since the CFL criteria for compressible flow is connected with the sound of speed.

### 2.2.2 Wet Steam Equation

All the simulations performed, use the Wet Steam model available in ANSYS Fluent. This software models the two-phase flow using Navier-Stokes equations, in addition to two transport equations which the liquid-phase mass fraction ($\beta$) and the number of liquid droplets per unit volume ($\eta$). For the Wet Steam model, only the density-based solver is available. There are some assumptions that need to be considered for this model:

- velocity slip between droplets and vapor-phase is negligible.
interactions between the droplets are neglected.

- mass fraction of the condensed phase, or liquid mass fraction, \( \beta \), is small: \( \beta < 0.2 \).
- as the size of droplets is tiny (0.1\(\mu\)m to 100\(\mu\)m), its volume is neglected.

Using the previous assumption the mixture density is connected with the vapor density:

\[
\rho = \frac{\rho_v}{1 - \beta}
\]  
(2.6)

Also, the pressure and temperature of the mixture will be the same as the pressure and temperature of the vapor-phase. However, the mixture properties which are related to liquid and vapor properties through the liquid mass fraction \( \beta \), use the following mixing law:

\[
\phi_m = \phi_l \beta + (1 - \beta) \phi_v
\]  
(2.7)

where \( \phi \) represents one of the following thermodynamic properties: \( h, s, C_p, C_v, \mu \) or \( k_t \).

The two transport equations that are added to solve condensation are the following:

- Governing the mass fraction of the condensed liquid phase (\( \beta \)):

\[
\frac{\partial \rho \beta}{\partial t} + \nabla \cdot (\rho \vec{v} \beta) = \Gamma
\]  
(2.8)

where \( \Gamma \) is the mass generation rate caused by condensation and evaporation (kg per unit volume per second).

- Modelling the evolution of the number density of the droplets per unit volume (\( \eta \)):

\[
\frac{\partial \rho \eta}{\partial t} + \nabla \cdot (\rho \vec{v} \eta) = \rho I
\]  
(2.9)

where \( I \) is the nucleation rate (number of new droplets per unit volume per second).

### 2.2.3 Phase Change Model

The phase change model assumes the next hypothesis:

- Condensation is homogeneous
Droplet growth is based on average representative mean radii

- Droplet is assumed to be spherical
- Droplets are surrounded by infinite vapor space
- Heat capacity of the fine droplets is negligible compared with the latent heat released in condensation.

The mass generation rate Γ in the classical nucleation theory during non-equilibrium condensation process is given by the sum of mass increase due to nucleation (the formation of critically sized droplets) and also due to the growth/demise of these droplets [18].

Therefore, Γ is written as:

$$\Gamma = \frac{4}{3} \pi \rho_l I r^3_\ast + 4\pi \rho_l \eta \frac{\partial \bar{r}}{\partial t}$$

(2.10)

where \(r\) is the average radius of the droplet and \(r_\ast\) is the Kelvin-Helmholtz critical droplet radius, above this value the droplet will grow and below, the droplet will evaporate.

The critical radius, \(r_\ast\), is given by [19]:

$$r_\ast = \frac{2\sigma}{\rho_l RT \ln S}$$

(2.11)

where \(\sigma\) is the liquid surface tension evaluated at the temperature \(T\), \(\rho_l\) is the condensed liquid density and \(S\) is the super saturation ratio defined as the ratio of vapor pressure to the equilibrium saturation pressure:

$$s = \frac{P}{P_{sat}(T)}$$

(2.12)

The expansion is usually fast, forcing not to move through equilibrium states and having values of supersaturation ratio \(S\) higher than one.

Condensation process implies two mechanisms, the mass transfer from the vapor to the droplets and the heat transfer between the droplets and the vapor as latent heat. This energy transference is presented by Young [19] and used by Ishazaki [18] and is
represented as the average radius change of the droplet:

\[
\frac{\partial r}{\partial t} = \frac{P}{h_{lv} \rho_l \sqrt{2 \pi R T}} \left( \frac{\gamma + 1}{2 \gamma} C_p (T_0 - T) \right)
\]  

(2.13)

where \( T_0 \) is the droplet temperature.

The classical theory of homogeneous nucleation describes the formation of the liquid as droplets of a supersaturated phase without impurities or irregular particles. The nucleation rate given by the classical theory of homogeneous nucleation [18] and corrected for non-isothermal effects is:

\[
I = q_c \left( \frac{\rho_l}{1 + \theta} \right) \frac{2 \sigma}{M_m \pi} e^{-\frac{4 \pi r^2 \sigma}{m \rho_l}}
\]  

(2.14)

where \( q_c \) is the evaporation coefficient, \( k_b \) is the Boltzmann, \( M_m \) is the mass of one molecule, \( \sigma \) is the liquid surface tension, and \( \rho_l \) is the liquid density at the temperature \( T \).

The non-isothermal correction factor, \( \theta \), is given by:

\[
\theta = \frac{2(\gamma - 1)}{\gamma + 1} \frac{h_{lv}}{RT} \left( \frac{h_{lv}}{RT} - 0.5 \right)
\]  

(2.15)

where \( h_{lv} \) is the specific enthalpy of evaporation at pressure \( P \) and \( \gamma \) is the ratio of specific heat capacities.

### 2.3 RANS Turbulence Models

In this thesis, turbulence modelling uses the RANS (Reynolds Averaged Navier-Stokes) models.

Turbulent flows are depicted by fluctuation of all transported variables. These fluctuations in velocity and the other variables might lead to fluctuations of energy and momentum. In the cases that fluctuations are of small scales and have high frequency, it is moderately expensive to compute. Alternately, the instantaneous governing equations can be ensemble averaged to eliminate these small scales, having a set of arranged equations that are less costly to resolve. The velocity field in this method is averaged over a time period which is much higher than time constant in velocity fluctuations.
These averaged equations are not closed (have more degrees of freedom) and they need a closure known as Boussinesq hypothesis [20], which provides its closure. Boussinesq hypothesis assumes that the turbulence is isotropic, which is a highly improbable if the flow is not simple enough. The k-$\varepsilon$, RNG-k-$\varepsilon$ and k-$\omega$ models are based on this approach.

There are different turbulence models in the RANS, and two of them are used in this thesis.

### 2.3.1 Standard k-$\varepsilon$ Model

The Standard k-$\varepsilon$ is probably the most used turbulence model as it is robust and well-validated. This model is only applicable for strongly turbulent flows since it is based on the assumption that the flow is fully turbulent (renders the effects of molecular viscosity negligible).

The Standard k-$\varepsilon$ model is based on model transport equations for the turbulence kinetic energy ($k$) and its dissipation rate ($\varepsilon$). The model transport equation for $k$ is derived from the exact equation, while the model transport equation for $\varepsilon$ was obtained using physical reasoning and bears little resemblance to its mathematically exact counterpart.

The turbulence kinetic energy, $k$, and its rate of dissipation, $\varepsilon$, are obtained from the following transport equations:

$$\frac{\partial}{\partial t} + \frac{\partial}{\partial x_i} (\rho k u_i) = \frac{\partial}{\partial x_j} \left[ \left( \mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right] + G_k + G_b - \rho \varepsilon - Y_M + S_k \quad (2.16)$$

Turbulent viscosity is obtained assuming its proportionality to the product between turbulent velocity scale and turbulent distance, and is given by:

$$\mu_t = \rho \cdot C_{\mu} \frac{k^2}{\varepsilon} \quad (2.17)$$

The effective viscosity of the fluid under a turbulent regime is given by:

$$\mu_{eff} = \mu + \mu_t \quad (2.18)$$

The model constants $C_{1\varepsilon}, C_{2\varepsilon}, C_{\mu}, \sigma_k,$ and $\sigma_\varepsilon$ have the following default values:

$$C_{1\varepsilon} = 1.44, C_{2\varepsilon} = 1.92, C_{\mu} = 0.09, \sigma_k = 1.0, \sigma_\varepsilon = 1.3 \quad (2.19)$$
Chapter 2. Physical and Numerical Model

This model is used in all cases considering that it is the most common. It is also suggested as slightly better by Hemidi et al. in [10] and Al-Ansary and Jeter [9].

2.3.2 Shear-Stress Transport k-ω Model

The SST (Shear-Stress Transport) k-ω model is a semi-empirical model based on transport equations for the turbulent kinetic energy ($k$) and the specific dissipation rate ($\omega$), which can also be thought of as the ratio of $k$ and $\varepsilon$ (from previous model).

The turbulence kinetic energy, $k$, and the specific dissipation rate, $\omega$, are obtained from the following transport equations:

$$
\frac{\partial}{\partial t}(\rho \cdot k) + \frac{\partial}{\partial x_i}(\rho \cdot k \cdot u_i) = \frac{\partial}{\partial x_j}(\Gamma_k \frac{\partial k}{\partial x_j}) + G_k - Y_k + S_k
$$

$$
\frac{\partial}{\partial t}(\rho \cdot \omega) + \frac{\partial}{\partial x_i}(\rho \cdot \omega \cdot u_i) = \frac{\partial}{\partial x_j}(\Gamma_\omega \frac{\partial \omega}{\partial x_j}) + G_\omega - Y_\omega + S_\omega
$$

The SST k-ω model has more accuracy for a wide range of flows such as transonic shock waves and adverse and adverse pressure gradient flows. It is usually applied when the flow is wall-resolved.

More information about turbulence models is found in the ANSYS Theory Guide [17].

2.4 Wall approach

Turbulent flows need some adaptations near the walls since the velocity is affected by the no-slip condition at the wall. Accurate solving near the walls leads to better prediction of the phenomena.

As seen in Figure 2.3, the near-wall region can be divided into three layers. The inner layer, ”viscous sublayer” is almost laminar. In the outer layer, ”fully-turbulent”, turbulence is more important. The intermediate region between the viscous and the fully-turbulent is the buffer layer, where effects of molecular viscosity and turbulence are equally important.
Chapter 2. Physical and Numerical Model

There are four different regions near the wall, depending on the $y^+$ which is a dimensionless variable to describe the $y$ distance to the wall.

$$y^+ = \frac{u_T \cdot y}{\nu}, \quad u^+ = \frac{1}{k} \ln y^+ + C^+$$

(2.22)

where $u_T$ is the friction velocity at the nearest wall, $y$ is the distance to the nearest wall and $\nu$ is the local kinematic viscosity.

There are two approaches to modelling the near-wall regions:

### 2.4.1 Wall Function

In one way, the viscosity-affected area is not resolved, and instead is solved by some wall functions, which are based on semi-empirical equations. This function connects the not resolved area with the fully turbulent region. In this case, high Reynolds turbulence models can be used.

For the $k-\varepsilon$ cases, the wall approach is set to Enhanced Wall Treatment for $\varepsilon$ equation. This treatment combines a two-layer model choosing the best in each case. That is to say, if the near-wall mesh is fine enough to be able to resolve the viscous sublayer (usually with the first node placed at $y^+ \sim 1$), the enhanced wall treatment is identical to the traditional two-layer model. If not, a blend of both models is applied.
2.4.2 Wall Resolved

The other option is to have a fully resolved region until the wall, including the viscous sublayer. In this case, the turbulence models applied need to be valid throughout the near-wall area.

2.5 Solver/Numerical setup

To perform the mathematical simulation, the finite volume method is used, in which the differential equations that control the physics are transformed to algebraic equations to be solved numerically.

The settings of the coupled solver are described below:

- Finite volume method: the governing equations are discretized using a control volume technique.
- Steady state.
- Discretization scheme: second order upwind.
- Central difference for viscous terms.
- Implicit (which allows having a Courant\(^1\) number bigger than 1).
- Roe flux splitting methodology.
- Gauss Siedel: The discretized system is solved in a coupled way with a Gauss-Siedel algorithm.

All simulations have been run with second order discretization scheme, although in some cases, the first part of the simulation was made with first order, and after converging, it was changed to second order for a higher accuracy in the final results. The same pattern was used after increasing the order for discretization for the Courant number, starting from a low value around 0.5 or 1, and successively increasing it when the simulation was approaching convergence.

\(^1\)The Courant number is an adimensional parameter which directly linked to the time step for the equation solving.
In most of the cases, the residuals easily decreased and became constant at values lower than $10^{-4}$. However, in other situations, the residuals did not decrease below $10^{-2}$ but the solution did not vary and the results were taken into account. Also, to check convergence the mass imbalance between inlet and outlet has been considered valid being lower than 5%.

Grid dependency tests were not performed regarding that the aim of this thesis was not to check if the grid was correct. As long as there is no experimental data to compare with, the purpose was to test the model and create a code for the properties. A mesh refinement near the wall was performed in one case to reach $y^+ \sim 1$ but the grid was so fine that the time cost to compute was not acceptable to keep along the thesis.
Chapter 3

Validation: Nozzle + Steam

3.1 Description of the cases

In the first place, as an approach to study condensation properties, the simulations in this chapter have been carried out with the built-in Wet Steam model. That is to say, that the water and steam properties have been used instead of refrigerant, which will be shown in Chapter 4.

The different cases are classified due to the inlet and outlet boundary conditions of the nozzle.

- Exp. No. 410-421 : To check the difference between these two cases performed by Moses and Stein [1].
- Turbulence : Compare a case with the two turbulence models.
- Shock Waves : Forcing to have shock waves inside the nozzle between the throat and the outlet.
- $\Delta T_{sat}$ : Modifying T in the inlet to see the evolution of condensation while reducing $\Delta T_{sat}$.

The aim of this variations is to check the differences between them, such as the condensation starting point or the behaviour of the pressure and temperature.
3.2 Numerical Setup

3.2.1 Mesh and Geometry

The nozzle used in the Chapters 3 and 4 corresponds to the one presented by Moses and Stein in [1] and later studied by L.Zori and F.Kelecy in [5].

The mesh is supplied by ANSYS Fluent and has 3000 quadrilateral cells. It only models half of the nozzle since it is symmetric. All the results are considered with the inlet located at 2 cm in the x-scale and the outlet is located at 14cm in the same scale. The nozzle shown in Figure 3.1 is double the original since symmetry has been applied.

As it is a 2D planar simulation, the depth of the nozzle is 1 m by default. In Figure 3.1 the inlet of the nozzle is the left limit of the nozzle. The symmetry axis divides the nozzle in two symmetric parts. The outlet is situated in the right limit of the nozzle.

The main measurements of the nozzle are grouped in the 3.1.

<table>
<thead>
<tr>
<th>Measures</th>
<th>cm</th>
</tr>
</thead>
<tbody>
<tr>
<td>Height inlet</td>
<td>2</td>
</tr>
<tr>
<td>Height outlet</td>
<td>0.7138</td>
</tr>
<tr>
<td>Total length</td>
<td>12</td>
</tr>
<tr>
<td>Length to throat</td>
<td>6.22</td>
</tr>
</tbody>
</table>

Table 3.1: Geometry measurement of the mesh (half of the real nozzle)

3.2.2 Boundary Conditions

The wet steam simulations are based on boundary conditions found in Table 3.2, which were experimentally tested by Moses and Stein [1]. The turbulence models $k-\varepsilon$ and $k-\omega$
are also compared in the first part of the results with the boundary conditions of Exp. No. 410 seen in Table 3.2.

<table>
<thead>
<tr>
<th>Experiment No.</th>
<th>$P_{inlet}$ [Pa]</th>
<th>$T_{inlet}$ [K]</th>
<th>$P_{outlet}$ [Pa]</th>
<th>$T_{outlet}$ [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>410</td>
<td>70727.5</td>
<td>377</td>
<td>5000</td>
<td>377</td>
</tr>
<tr>
<td>421</td>
<td>66807.8</td>
<td>385</td>
<td>5000</td>
<td>385</td>
</tr>
</tbody>
</table>

Table 3.2: Boundary conditions given by [1] and [5]

In the second part of the results, the boundary conditions are modified varying the superheating level of the inlet in the case of $\Delta T_{sat}$ evaluation, whereas in the shock wave analysis, it is the back pressure which is modified. The boundary conditions in the shock wave cases were checked in order to have one downstream the throat and upstream the outlet and can be checked in Table 3.3. In the cases where there is turbulence, Turbulent Intensity is imposed to be 5% and Hydraulic Diameter is $D_{h-inlet} = 0.03922m$ and $D_{h-outlet} = 0.01417m$.

<table>
<thead>
<tr>
<th>$P_{inlet}$ [Pa]</th>
<th>$T_{inlet}$ [K]</th>
<th>$P_{outlet}$ [Pa]</th>
<th>$T_{outlet}$ [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>70727.5</td>
<td>377</td>
<td>51995.7</td>
<td>377</td>
</tr>
</tbody>
</table>

Table 3.3: Boundary conditions for shock waves - WS

3.3 Results I

This section is thought to be the description of the results related with the boundary conditions. Experimental available results will be compared with the simulations.

3.3.1 Experiments No. 410-421

The steam expands isentropically from the inlet until the throat In Figure 3.2, the expansion along the nozzle is seen as the pressure ratio decreases along the nozzle. The main characteristic to point out is the pressure jump around $x = 0.11cm$. This occurrence is because of the release of vaporisation heat just in the moment that condensation starts to be significant. Also, this motivates a deviation in the thermodynamic state.

Liquid Mass Fraction plot in Figure 3.3 shows us the appearance of liquid droplets, when the value begins to be increase after $x = 0.1cm$. 
Figure 3.2: Pressure ratio - Exp. 410 and 421

Figure 3.3: Liquid Mass Fraction Plot - Exp. 410 and 421
Heat release is also reflected in Figure 3.4, in the point where the temperature increases some distance after the condensation appearance.

In Figure 3.4, at around $5 \sim 6$ cm in the x-scale, the values of the $\Delta T_{sat}$ begin to be negative as the state goes into the supersaturation region ($S$ higher than 1). One of the relevant issues of $\Delta T_{sat}$ when using steam is that the temperature has to be around $-37.5$ K and $-38.5$ K for the nucleation process to start, respectively for 410 and 421 experiments. This minimum $\Delta T_{sat}$ value, also called Wilson point, is reached around 1.5-2.5 cm downstream the throat (the throat is located at 8.22 cm in the x-scale).

The very beginning of the condensation is shown by the $\log_{10}$(Droplet Nucleation Rate) in Figure 3.5, which starts being perceptible a little before the throat, in the model case of boundary conditions 410. This does not imply directly the appearance of droplets but the starting of the creation.

As it can be seen in Figure 3.6, the differences between inviscid simulation and experimental results supplied by Moses and Stein [1] are really slight.

In Figure 3.7, there is the T-s plot for the Wet, Dry and Perfect gas cases. The perfect gas complies with a isentropic expansion, whereas the Wet and Dry cases do not. Also,
the Wet steam case has an increase of entropy and temperature in the point where condensation starts.

The mass flow rates for both experimental cases are showed in Table 3.4.

Figure 3.8 shows the static pressure along the nozzle. After the throat the pressure
Chapter 3. Validation: Nozzle + Steam

Figure 3.7: T-s plots with Wet, Dry and Perfect gas cases

(a) T-s

(b) T-s Zoom In

Table 3.4: Mass flow rate values in Exp. No. 410-421 cases - WS

<table>
<thead>
<tr>
<th>Case [Pa]</th>
<th>Mass flow rate, ( \dot{m} ) [kg/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exp. 410</td>
<td>0.5691</td>
</tr>
<tr>
<td>Exp. 421</td>
<td>0.5316</td>
</tr>
</tbody>
</table>

decrees more smoothly due to the appearance of condensation seen in the plot. Figure 3.9 illustrates the droplet nucleation rate, which highly increases after the throat and when condensation is quite strong disappears completely, as then, the liquid mass fraction increase is due to droplet growth.

Figure 3.8: Contour Static Pressure [Pa] - Exp. 410
3.3.2 Turbulence

In this case, both the $k$-$\varepsilon$ with EWT and $k$-$\omega$ SST were simulated. As it can be seen in 3.10, no great differences can be appreciated, between the turbulence models or even with the Inviscid and Laminar cases. It is worth mentioning that the mesh was modified in the $k$-$\omega$ SST model in order to have it wall resolved ($y^+ \approx 1$ near the wall).

As a relevant point it can be said that the differences between the cases might not be, that steam has quite an ideal behaviour and this is why differences are not being showed.

The different mass flow rates are shown in Table 3.5.
3.4 Results II

This section describes the results of the simulations for the different boundary conditions: \(\Delta T_{\text{sat}}\) variations to see the different levels of condensation and shock waves inside the nozzle to see its influence.

### 3.4.1 \(\Delta T_{\text{saturation}}\) variations

The aim at the beginning of this section was to try a wide range of inlet conditions, including saturated steam. However, with the boundary conditions below \(\Delta T_{\text{sat}} = 5\ \text{K}\), the simulations were really difficult to converge. This problems appeared due to the liquid mass fraction reached the limit of 0.2, set by the software. The scaled residuals, in this case, oscillate around \(10^{-1}\) and \(10^{-3}\), and these are not satisfactory enough according to the values set in Chapter 2.

The conditions that were applied finally go from 13.5 K, 10 K and 5 K above the saturation curve.
The most significant and obvious result obtained from this results, is that the closer to saturation the inlet condition is, the sooner the condensation starts. It can be checked in the plot in Figure 3.12. The profile of $\Delta T_{sat}$ in Figure 3.14 shows the profiles for three cases 5 K, 10 K and 13.5 K of supersaturation. Each inlet condition is different.
but at the outlet, as long as there is condensation the temperature is the same in the three cases.

Figure 3.14: $\Delta T_{\text{sat}}$ of inviscid cases - $\Delta T_{\text{sat}}$

The mass flow rates of the different inlet conditions are shown in Table 3.6. What would be expected is that the lower temperature in the inlet the higher mass flow rate, as the density should be higher. However, the 5 K case does not agree with this statement.

<table>
<thead>
<tr>
<th>Case [Pa]</th>
<th>Mass flow rate, $\dot{m}$ [kg/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>5 K</td>
<td>0.5642</td>
</tr>
<tr>
<td>10 K</td>
<td>0.5691</td>
</tr>
<tr>
<td>13.5 K</td>
<td>0.5316</td>
</tr>
</tbody>
</table>

Table 3.6: Mass flow rate values in $\Delta T_{\text{sat}}$ cases - WS

3.4.2 Shock Waves

In Figure 3.15a the differences between k-ε and k-ω can be appreciated. The shock wave happens sooner in the k-ω model than in the k-ε. However, the only case which is affected by condensation is the wet k-ε since there is a jump after $x = 0.1cm$.

Also, it can be seen that there is not much difference between the dry and wet cases for k-ω. This is due to the low strength of the condensation as it can be seen in the 3.16a.
Even $k$-$\epsilon$ has a more important liquid appearance than $k$-$\omega$, it is a low condensation level compared with the one seen in Section 3.3.

\begin{figure}[h]
\centering
\begin{subfigure}{0.45\textwidth}
\includegraphics[width=\textwidth]{pressure_ratio.png}
\caption{Pressure ratio}
\end{subfigure}
\hfill
\begin{subfigure}{0.45\textwidth}
\includegraphics[width=\textwidth]{delta_T_sat.png}
\caption{$\Delta T_{sat}$}
\end{subfigure}
\caption{Pressure Ratio and $\Delta T_{sat}$ - SW}
\end{figure}

\begin{figure}[h]
\centering
\begin{subfigure}{0.45\textwidth}
\includegraphics[width=\textwidth]{liquid_mass_fraction.png}
\caption{Liquid Mass Fraction}
\end{subfigure}
\hfill
\begin{subfigure}{0.45\textwidth}
\includegraphics[width=\textwidth]{log10_dnr.png}
\caption{Log10(Droplet Nucleation Rate)}
\end{subfigure}
\caption{LMF and Log10(DNR) - SW}
\end{figure}

In Figure 3.16b, it is clearly seen where is located the shock wave in the $k$-$\omega$ model, since the Liquid Mass Fraction decreases abruptly.

The location of the shock wave in the wet cases is the following:

- Wet $k$-$\epsilon$ shock wave is located at $x = 11.70$ cm
- Wet $k$-$\omega$ shock wave is located at $x = 9.86$ cm
Pressure ratio is seen in as a contour plot in Figure 3.17. The shock wave is seen downstream the throat when pressure changes abruptly.

Both contour plots, Figure 3.18 and 3.19 show the condensation appearance and the liquid presence around the shock wave.

<table>
<thead>
<tr>
<th>Case</th>
<th>Mass flow rate, $\dot{m}$ [kg/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wet k-$\varepsilon$</td>
<td>0.5600</td>
</tr>
<tr>
<td>Dry k-$\varepsilon$</td>
<td>0.5600</td>
</tr>
<tr>
<td>Wet k-$\omega$</td>
<td>0.5579</td>
</tr>
<tr>
<td>Dry k-$\omega$</td>
<td>0.5579</td>
</tr>
</tbody>
</table>

Table 3.7: Mass flow rate values in Shock Wave cases - Wet Steam
Figure 3.19: Liquid Mass Fraction [-] contour plot, Wet k-ε - SW

The flow between the wet and dry cases does not vary due to the low strength of condensation. It would be expected to find a difference between the dry and wet cases due to the different density of liquid phase.
Chapter 4

Validation: Nozzle + Refrigerant

4.1 Description of the cases

The cases that will be studied in this Chapter have R134a as the working fluid. The cases are divided in two parts. In the first part, the conditions are set to be fully supersonic (FS), that is Ma=1 in the throat and Mach>1 downstream. In the second part, a shock wave is forced to happen downstream of the throat and before the nozzle outlet. The Boundary conditions are described carefully in section 4.2.2. For both cases, the simulations are made for wet steam and dry steam, deactivating condensation (considering steam as refrigerant vapor from now on).

Little and Garimella suggest in [15] that in conventional operation of a refrigeration cycle the state in the entrance of the ejector, thus the nozzle, should be superheated at around 10 K. This works as a measure to protect the moving parts in case of using rotating machinery. Anyway, the conditions will be varied downwards to study the consequences of condensation in the motive flow of the ejector, which goes through the nozzle.

4.1.1 Refrigerant

Water, used in Chapter 3, offers high heat of vaporization and has no environmental impact. It could be a perfect refrigerant if it was not for the limit in cooling temperature (only above 0°C). Also, water has large specific volume and the cycle pipes diameter should be too big. In contrast with the previous chapter, the working fluid here is R134a.
This refrigerant is a low temperature halocarbon which offers the advantage of low GWP and no ODP compared with other halocarbons. As Chunnanond [2] suggests, the halocarbon refrigerants provide larger performance and lower temperature requirements for the heat recovery than water.

4.2 Numerical Setup

4.2.1 Geometry and Mesh

The geometry and mesh is the same that is described in Chapter 3.

4.2.2 Boundary Conditions

As previously seen, the cases that will be studied will be divided in four parts:

- Fully supersonic
  - Dry Steam
  - Wet Steam
- Shock wave
  - Dry Steam
  - Wet Steam

The boundary conditions used along this Chapter were set up regarding the recommendations suggested by Little and Garimella [15]. It is known that there should be a level of $\sim 10K$ of superheat. Anyway, the aim was to find condensation and the first simulations showed that at this superheat level did not provide any liquid appearance. The conditions for the Fully Supersonic cases are the ones seen in 4.1. The cases with $\Delta T_{sat} \leq -4$ K had a condensation level higher than the limit of the software ($\beta = 0.2$), that is why the inlet boundary conditions do not go below that point.

The second part of the simulations were made in order to see the behaviour having shock waves and condensation (if possible) in the nozzle, the back pressure is increased as seen
Table 4.1: Boundary conditions for fully supersonic cases - R134a

<table>
<thead>
<tr>
<th>Case</th>
<th>$P_{\text{inlet}}$ [Pa]</th>
<th>$T_{\text{inlet}}$ [K]</th>
<th>$P_{\text{outlet}}$ [Pa]</th>
<th>$T_{\text{outlet}}$ [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta T_{\text{sat}}$ = 0 K</td>
<td>3040500</td>
<td>360</td>
<td>600000</td>
<td>360</td>
</tr>
<tr>
<td>$\Delta T_{\text{sat}}$ = -2 K</td>
<td>3040500</td>
<td>358</td>
<td>600000</td>
<td>358</td>
</tr>
<tr>
<td>$\Delta T_{\text{sat}}$ = -3 K</td>
<td>3040500</td>
<td>357</td>
<td>600000</td>
<td>357</td>
</tr>
</tbody>
</table>

Table 4.2: Boundary conditions for shock wave cases - R134a

<table>
<thead>
<tr>
<th>Case</th>
<th>$P_{\text{inlet}}$ [Pa]</th>
<th>$T_{\text{inlet}}$ [K]</th>
<th>$P_{\text{outlet}}$ [Pa]</th>
<th>$T_{\text{outlet}}$ [K]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta T_{\text{sat}}$ = -2 K</td>
<td>3040500</td>
<td>358</td>
<td>2500000</td>
<td>358</td>
</tr>
<tr>
<td>$\Delta T_{\text{sat}}$ = -3 K</td>
<td>3040500</td>
<td>357</td>
<td>2500000</td>
<td>357</td>
</tr>
<tr>
<td>$\Delta T_{\text{sat}}$ = -4 K</td>
<td>3040500</td>
<td>356</td>
<td>2500000</td>
<td>356</td>
</tr>
</tbody>
</table>

in 4.2. Similarly to the fully supersonic conditions, in the shock waves cases the inlet conditions could not be below $\Delta T_{\text{sat}}$ = -3K as, then the

All the cases are simulated with the k-$\varepsilon$ turbulence model. Turbulent Intensity is imposed to be 5% for the inlet flow and outlet backflow and the Hydraulic Diameter is $D_{h-\text{inlet}}$ = 0.03992 m and $D_{h-\text{outlet}}$ = 0.01417m.

### 4.2.3 R134a User Defined Functions

The difference of this chapter compared with Chapter 2 is that in this case, the properties of the fluid were introduced as User Defined Functions (UDF). UDF that need to be implemented are separated in four groups and are the following:

- **Saturation Properties:**
  - saturation pressure, $P_{\text{sat}}$, as a function of $T$;
  - saturation temperature, $T_{\text{sat}}$, as a function of $P$ and a starting guess $T$;

- **Equation of State Properties:**
  - Equation Of State pressure, $P$, as a function of vapor density $\rho_v$ and $T$;
  - Equation Of State vapor density, $\rho_v$, as a function of $P$ and $T$;

- **Vapor Properties:**
  - vapor specific heat at constant pressure, $C_{p-v}$, as a function of vapor density $\rho_v$ and $T$;
validating nozzle + refrigerant

- vapor specific heat at constant volume, $C_{v-v}$, as a function of vapor density $\rho_v$ and T;
- vapor specific enthalpy, $h_v$, as a function of vapor density $\rho_v$ and T;
- vapor specific entropy, $s_v$, as a function of vapor density $\rho_v$ and T;
- vapor dynamic viscosity, $\mu_v$, as a function of vapor density $\rho_v$ and T;
- vapor thermal conductivity, $k_{tv}$, as function of vapor density $\rho_v$ and T;

• Liquid Properties:
  - saturated liquid density, $\rho_l$, as function of T;
  - saturated liquid specific heat at constant pressure, $C_{p-l}$, as function of T;
  - liquid dynamic viscosity, $\mu_l$, as function of T;
  - liquid thermal conductivity, $k_{tl}$, as function of T;
  - liquid surface tension, $\sigma$, as function of T;

The three different codes that were designed can be found in the Appendix B.

4.2.3.1 Ideal Gas Properties

The aim of the Ideal Gas UDF was to simplify the functions implemented at most, in order to prevent errors. That is why most of the equations/functions were idealized and reduced.

Saturation Properties: For Wet Steam, the saturation pressure and temperature were extracted from [21] and it was checked in case R134a was also described. This refrigerant was not available there, but was useful to understand the vapor properties pattern.

In the code for R134a, saturation pressure function is obtained from NIST REFPROP [22]. The values of pressure and temperature are from the vapor-liquid saturation curve. Pressure as a function of temperature is obtained using Microsoft Excel as a polynomial regression of 6th grade.

In order to avoid loops due to not enough accuracy, the saturation temperature function is based on an iterative method given by ANSYS Fluent User’s Guide [4].
Equation of State Properties: As ideal gas, the pressure, $P$, and vapor density, $\rho_v$, are both given by the Ideal Gas equation:

\begin{align*}
P &= \rho_v \cdot R_{\text{gas-v}} \cdot T \\
\rho_v &= \frac{P}{R_{\text{gas-v}} \cdot T}
\end{align*}

where $T$ is the temperature and $R_{\text{gas-v}}$ is the vapor gas constant equal to 81.490367 J/kg·K.

Vapor Properties:

Vapor specific heat at constant pressure, $C_{p-v}$, is built in a way which makes the entropy as close as real enthalpy. Initially, $C_{p-v}$ was a constant value, but the ideal gas enthalpy was affecting negatively the simulations, having convergence problems.

Vapor specific heat at constant volume, $C_{v-v}$, complies the ideal gas equation:

\begin{equation}
R = C_{p-v} - C_{v-v}
\end{equation}

Enthalpy, $h$, complies with its ideal gas equation:

\begin{equation}
h = T \cdot C_{p-v}
\end{equation}

Entropy, $s$, complies with the ideal gas equation:

\begin{equation}
s = C_{p-v} \cdot \ln\left(\frac{T}{T_0}\right) + R_{\text{gas-v}} \cdot \ln\left(\frac{P}{P_{\text{gas-v}}}ight)
\end{equation}

Vapor dynamic viscosity, $\mu_v$, is a constant value based on the same structure as the UDWSF suggested in [4] p. 1424. It is a number close to the state in the inlet.

Vapor thermal conductivity, $k_{tv}$, is also a constant value based on the same structure as the UDWSF suggested in [4] p. 1424. It is a number close to the state state in the inlet.

Liquid Properties:
The liquid properties are all based on the polynomial regression of the values in a saturated state from the NIST REFPROP library [22], as the liquid does not change too much.

4.2.3.2 Real Gas Properties

Saturation Properties:

The saturation properties for the real gas are the same as in the ideal, since the state values come from the NIST REFPROP library [22] which contain real gas state.

Equation of State Properties:

To decide which Equation Of State was going to be introduced in the code, the basis was to have an EOS as simple as possible. After having a look in other works, the decision was made between Redlich-Kwong and Peng-Robinson equations. In Figure 4.1, three different EOS are plotted for two constant T (340 and 365 K). For similar values at the inlet boundary conditions the error compared with real gas is less than 1% for Peng-Robinson, around 3% for Redlich-Kwong and around 40% for Ideal Gas.

Due its low deviation from real properties near the conditions that will be used, the EOS that is chosen is the Peng-Robinson [23] and it can be written as:

\[ P = \frac{RT}{v - b} - \frac{a\alpha}{v^2 + 2\nu b - b^2} \] (4.6)
where \( a = a(T_c) \), \( b = b(T_c) \), \( \alpha \), \( T_r \), and \( k \) are:

\[
a = \frac{0.457235R^2T_c^2}{P_c} \quad (4.7)
\]

\[
b = \frac{0.07796RT_c}{P_c} \quad (4.8)
\]

\[
\alpha = (1 + k(1 - \sqrt{T_r}))^2 \quad (4.9)
\]

\[
T_r = \frac{T}{T_c} \quad (4.10)
\]

\[
k = 0.37464 + 1.54226 \cdot \omega - 0.26992 \cdot \omega^2 \quad (4.11)
\]

To implement the \( P \) function, all the constants above-mentioned need to be defined. But for the \( v \) equation, there is:

\[
v^3 + v^2(b - \frac{RT}{P}) + v(-3b^2 - \frac{2RTb}{P} + \frac{\alpha a}{P}) + (b^3 + \frac{RTb^2}{P} - \frac{\alpha ab}{P}) = 0 \quad (4.12)
\]

In order to solve \( v \) (and density, \( \rho = 1/v \)), this cubic equation has to be studied carefully. To understand the procedure that is considered in the real gas code, Figure 4.2 illustrates a Pressure - Volume plot for a general cubic EOS for volume.

![Figure 4.2: Qualitative P - v Plot for cubic EOS from [4]](image)

There is a subcritical isotherm ABFED at a temperature \( T \). The curve C1CC2 is the critical isotherm and ACD is the saturation dome. The three roots that solve the EOS at a saturation Pressure \( P_s \) are the points A, F, and D: A represents the molar volume of the saturated liquid, D represents the molar volume of the saturated vapor, and F does not have any physical meaning. BCE is the spinoidal curve, the boundary beyond
the EOS is not valid as the local derivative of pressure with respect to volume becomes positive.

Therefore, the root that it is "correct" is the one on the right. In order to solve \( v \), the Newton-Raphson iterative pattern is implemented in this way:

Given the \( f(v) \) function, its derivative \( f'(v) \) and a first guess \( v_0 \), which is the maximum \( v \) value acceptable \( (v = 1m^3/kg) \), there is a first approximation for the root:

\[
v_1 = v_0 - \frac{f(v_0)}{f'(v_0)} \tag{4.13}
\]

It means that the point \((v_1, 0)\) is the intersection with the x axis of the tangent of the function \( f \) at the point \((x_0, f(x_0))\). This process is performed successively until the variation of the \( x_n \) compared with the previous \( x_{n-1} \) is lower than a set limit. The code for this can be checked in Appendix B in the Real Gas section.

**Vapor Properties:**

Vapor properties functions are built out of the properties supplied in REFPROP NIST [22]. The saturation properties together with the vapor properties for a range of pressure from 3.5 MPa to 50 kPa were used to create multiple linear regression functions with \( \rho \) and \( T \) as input variables. The pattern for obtaining the functions was: including all input variables \( T, \rho, T^2, \rho_2 \) and \( T \cdot \rho \). Perform a multiple linear regression, and check the p-values of the variables. If a p-values is higher than 0.005 the corresponding variable is removed from the model. After, perform another multiple linear regression and keep the same criteria up to a point where all the p-values are lower than 0.05. Find the equations of the vapor properties in Appendix A.

**Liquid Properties:**

The equations for liquid properties are the same as in ideal gas.

### 4.3 Results

Simulations with Ideal Gas came out properly until \( \Delta T_{sat} = -3 \). After that point, the maximum limit for the liquid mass fraction \( (\beta = 0.2) \) was reached, and the software
does not guarantee correct values. Also, due to this limit of the model, the simulations are impossible to converge.

The main problem appeared with the Real Gas code, when at the very beginning of the simulations, divergence problems turned out. Multiple solver adaptations were made to avoid divergence:

- Courant number was reduced to 0.1
- Discretization schemes were set to first order upwind
- Testing more favorable initialization values, and even with data from the perfect gas simulation solution

Any of the adjustments worked, then the idea was to change the code starting with the real gas properties. Trying to simplify and change the real gas properties did not turn into any improvements, therefore, the next step was testing a new code with the perfect gas EOS and real gas properties to see which and where was the problem. This code worked almost as smoothly as the Perfect Gas code in terms of convergence. Below, this mixed code is called Hybrid Gas (HG).

The results below are based only in the Perfect Gas and Hybrid Gas cases.

### 4.3.1 Fully supersonic

It can be checked if the flow is indeed fully supersonic by looking at the Mach number in Figure 4.3, as Mach 1 is reached in the throat and then increases. In all cases, the pattern for the Mach number is the same, only zooming in in the final the nozzle it is seen a different behaviour for the wet case for $\Delta T_{sat} = -3 \text{ K}$.

The same structure is seen in Pressure ratio plots in Figure 4.4. Pressure ratio decreases until a point where the wet case for $\Delta T_{sat} = -3K$ the pressure differs as it keeps slightly higher values (Figure 4.4b).

This similar performance for velocity and pressure in all the cases is because almost none of them have condensation inside the nozzle. The liquid appearance in the two cases where there is actually condensation can be appreciated in the Liquid Mass Fraction...
plots in Figure 4.5. As it can be seen, condensation is only noticeable in the \( \Delta T_{\text{sat}} = -3 \) K, even though that in \( \Delta T_{\text{sat}} = -2 \) K there is a start.

The strength of the condensation could be almost the same as the Droplet nucleation ratio which gives an idea of the number of new droplets per unit volume and second. In Figure 4.6, it is seen that the droplet appearance happens before in the -3K case, as it is obviously inner in the saturation dome.

In order to check the level of saturation, the \( \Delta T_{\text{sat}} \) plot in Figure 4.7 shows a different
Chapter 4. Validation: Nozzle + Refrigerant

Figure 4.5: Liquid Mass Fraction plots for FS cases - PG

Figure 4.6: Log10(Droplet Nucleation Ratio) for Fully Supersonic cases - PG
pattern compared with the wet steam cases. First $\Delta t_{\text{sat}}$ increases slightly and after the throat it decreases more rapidly. In the case of $\Delta T_{\text{sat}} = -3$ K, the minimum value of $\Delta T_{\text{sat}}$ is $-11.6$ K, below that point condensation has an enough important level and there is a temperature jump due to the heat release. In the case of $\Delta T_{\text{sat}} = -2$, where there is a little condensation, the minimum value is reached on the outlet and is $-11.5$ K.

![Figure 4.7: $\Delta T_{\text{sat}}$ for FS cases - PG](image)

However, in the Hybrid gas cases there is a quite different pattern also for the $\Delta T_{\text{sat}}$ as seen in Figure 4.8. The decrease during the expansion is less pronounced compared with the Perfect Gas code results. This might be due to the different equations for the $C_{p-v}$, $C_{v-v}$ and $k_v$.

Anyway, in the Hybrid gas there are no available results with condensation because at $\Delta T_{\text{sat}} = -3$ K there was condensation but it was too strong for the software to support ($\beta$ was greater than 0.2 as it can be seen in the Figure 4.9). For Perfect gas cases, the limit was reached in the values below $\Delta T_{\text{sat}} = -4$ K.

Regarding the mass flow rates in Table 4.3, in case $\Delta T_{\text{sat}} = 0$ K, there is no condensation at all so the mass flow rate is only specified once. In the case where $\Delta T_{\text{sat}}$ is $-2$ K, as previously seen, the liquid fraction is so weak that the mass flow rate does not change.
at all although the wet model is activated. However, from $\Delta T_{sat} = -3$ K, mass flow rate does not change, this could be due to the outlet proximity of the condensation onset.

In addition, the increase of the mass flow rate (seen in Table 4.3) with the lowering of the inlet temperature might be explained by the increase of the density value. That is, the temperature reduction leads to a higher density and if the volume of flow is kept, the mass flow rate is larger.
Table 4.3: Mass flow rate values in fully supersonic Perfect and Hybrid cases - R134a

<table>
<thead>
<tr>
<th>Case</th>
<th>Mass flow rate (P), $\dot{m}$ [kg/s]</th>
<th>Mass flow rate (H), $\dot{m}$ [kg/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta T = 0$ K</td>
<td>57.9697</td>
<td>58.31175</td>
</tr>
<tr>
<td>Dry $\Delta T = -2$ K</td>
<td>58.1668</td>
<td>58.50657</td>
</tr>
<tr>
<td>Wet $\Delta T = -2$ K</td>
<td>58.1668</td>
<td>58.50657</td>
</tr>
<tr>
<td>Dry $\Delta T = -3$ K</td>
<td>58.2660</td>
<td>58.60457</td>
</tr>
<tr>
<td>Wet $\Delta T = -3$ K</td>
<td>58.2660</td>
<td>$^1$</td>
</tr>
</tbody>
</table>

4.3.2 Shock waves

In the Perfect Gas environment it is impossible to find condensation within the limits of the software. Cases up to $\Delta T_{sat} = -4$ K have resulted into no condensation at all, and below that condition the condensation is too strong and the liquid mass fraction ($\beta$) is higher than 0.2.

Using the Hybrid Gas code, the cases below $\Delta T_{sat} = -2$ K become also into high condensation values. The case $\Delta T_{sat} = -3$ K showed in the Figures below is the dry case, in order to have some data to compare the values of the PG cases. In Figure 4.12, the Liquid Mass Fraction for the HG wet $\Delta T_{sat} = -3$ K case is showed. It should be taken into account that this solution is not converged and it works as an example to understand what happens with Liquid Mass Fraction reaching the limitations.

Both Mach number and Pressure ratio are shown in Figure 4.10 and 4.11, comparing the results of Perfect Gas and Hybrid gas. It is illustrated that there is not much difference regarding the pressure and the velocity for this cases.

In Figure 4.13, the $\Delta T_{sat}$ is illustrated for the different case available. Similarly as seen in the fully supersonic cases, the pattern for the Hybrid Gas cases is quite different from the Perfect Gas. In this case, as already told, it could be predicted that condensation would start first in the Hybrid Cases due to the more rapid decrease of temperature inside the saturation dome.

Regarding the mass flow rate, it is seen that for hybrid cases is a little higher than when using perfect gas code. As previously seen, if the input temperature is lower the mass flow rate increases.
Chapter 4. Validation: Nozzle + Refrigerant

Figure 4.10: Pressure ratio for SW cases - R134a

Figure 4.11: Mach number for SW cases - R134a

<table>
<thead>
<tr>
<th>Case</th>
<th>Mass flow rate (P), $\dot{m}$ [kg/s]</th>
<th>Mass flow rate (H), $\dot{m}$ [kg/s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\Delta T_{sat} = -2$ K</td>
<td>58.1666</td>
<td>58.5026</td>
</tr>
<tr>
<td>$\Delta T_{sat} = -3$ K</td>
<td>58.2660</td>
<td>58.6046 (only dry)</td>
</tr>
<tr>
<td>$\Delta T_{sat} = -4$ K</td>
<td>58.3656</td>
<td></td>
</tr>
</tbody>
</table>

Table 4.4: Mass flow rate values in SW PG and HG
Figure 4.12: Contour plot for Liquid Mass Fraction [-] for $\Delta T_{\text{sat}} = -3$ K SW - HG

Figure 4.13: $\Delta T_{\text{sat}}$ plot for SW cases
Chapter 5

Conclusions

5.1 Conclusions

Wet Steam built-in model was studied and tested for different conditions. The results have been useful to see how the model worked and how the properties were introduced in the UDWSPF.

For the R134a, three codes have been developed: ideal gas, hybrid gas (ideal EOS and real properties) and real gas. The simulations for ideal gas and hybrid gas, as the real gas cases were had divergence problems related to the AMG solver. The results for ideal and hybrid gas have been studied and compared between them. There are many differences that depend too much on the code, so it should be further studied and developed in order to approximate the results as much as possible to the reality.

In addition, many of the cases did not have condensation since the software has a limitation for liquid mass fraction that has restricted a lot the range of boundary conditions. This situation has been an important disadvantage that was not expected from the beginning.

Experimental tests are needed in order to validate the CFD results in this case, otherwise the results can only be compared with different models. This way, for example the most appropriate turbulence model could be chosen in order to continue the simulations.
5.2 Futur Work

Some pending work has been introduced during the thesis, however, the summary of the ideas that should be developed and more studied are suggested below:

- Try to use the codes that work (perfect and hybrid) in a full ejector and study the results.
- Go in depth in the mesh resolution. It was not in the objectives of the thesis but it is really important.
- Study exhaustively the code for real gas and correct it:
  - Modify the Ideal Gas EOS changing R value (which is the slope of the line i.e. for P-ρ for a fixed T) to make it closer to the real gas states.
  - Find a better and simpler equation of state to implement in the real gas code.
- Verify that the code works for different working fluids.
- Study in depth the different turbulence models, and compare the different calculation to experimental results.
Appendix A

Property functions for R134a

A.1 Saturation pressure function

\[ P_{\text{sat}} = 1.49560 \times 10^7 - 3.64411 \times 10^5 \cdot T + 3.61522 \times 10^3 \cdot T^2 - 1.86330 \times 10 \cdot T^3 \]
\[ + 5.25909 \times 10^{-2} \cdot T^4 - 7.78881 \times 10^{-5} \cdot T^5 + 4.94851 \times 10^{-8} \cdot T^6 \] (A.1)

A.2 Vapor property functions for real gas

A.2.1 \( C_{p-v} \)

\[ C_{p-v} = -4.42590 \times 10^2 + 7.36158 \cdot T - 8.68716 \times 10^{-3} \cdot T^2 - 6.22141 \cdot \rho + 6.98903 \times 10^{-2} \cdot \rho^2 \] (A.2)

A.2.2 \( C_{v-v} \)

\[ C_{v-v} = 2.14813 \times 10^2 + 1.90981 \cdot T - 3.42800 \times 10^{-4} \cdot T^2 \]
\[ + 5.02810 \cdot \rho - 5.52774 \times 10^{-4} \cdot \rho^2 - 1.06197 \times 10^{-2} \cdot T \cdot \rho \] (A.3)

A.2.3 \( h_v \)

\[ h_v = 1.51022 \times 10^5 + 9.28061 \times 10^2 \cdot T - 2.74230 \times 10^2 \cdot \rho \] (A.4)
A.2.4 \( s_v \)

\[
 s_v = 1.82994 \times 10^3 - 1.07844 \cdot T + 4.62066 \cdot 10^{-3} \cdot T^2 - 6.35613 \cdot \rho + 1.30475 \cdot 10^{-2} \cdot \rho^2 + 5.10279 \cdot 10^{-3} \cdot T \cdot \rho
\]  

(A.5)

A.2.5 \( mu_v \)

\[
 mu_v = -9.62866 \cdot 10^{-8} + 4.13590 \cdot 10^{-8} \cdot T - 4.63950 \cdot 10^{-12} \cdot T^2

- 2.85153 \cdot 10^{-8} \cdot \rho + 7.33865 \cdot 10^{-11} \cdot \rho^2 + 8.16447 \cdot 10^{-11} \cdot T \cdot \rho
\]  

(A.6)

A.2.6 \( k_{tv} \)

\[
 k_{tv} = -9.30258 \cdot 10^{-3} + 7.09679 \cdot 10^{-5} \cdot T + 1.65925 \cdot 10^{-8} \cdot T^2

+ 5.59051 \cdot 10^{-5} \cdot \rho + 2.42633 \cdot 10^{-7} \cdot \rho^2 - 1.66511 \cdot 10^{-7} \cdot T \cdot \rho
\]  

(A.7)
Appendix B

User Defined Functions for R134a

B.1 UDF for ideal gas

B.1.1 Ideal Gas EOS - Ideal Gas Properties

/**********************************************************************/
/* User Defined R134a Properties:                                    
EOS: Ideal Gas Eq.                                                    
Perfect Gas Properties                                               
Real Liquid Properties                                               
Author: Marina Gallach Palma                                         
Date : 7 July 2015                                                   */
/***********************************************************************/

#include "udf.h"
#include "stdio.h"
#include "ctype.h"
#include "stdarg.h"

/*Global Constants for this model*/

real ws_TPP=169.85;
real ws_aaa=0.01;
real cpg=585.66;
/* Cp-vapor at low-pressure region*/

DEFINE_ON_DEMAND(I_do_nothing)
{ /* This is a dummy function to allow us to use */
  /* the Compiled UDFs utility*/
}

void wetst_init(Domain *domain)
{
  /* You must initialize these material property constants.. they will be used in the wet steam model in fluent */
  ws_Tc = 374.21; /*Critical Temp. */
  ws_Pc = 4059300.0; /*Critical Pressure */
  mw_f = 102.03; /*fluid droplet molecular weight (water) */
  Rgas_v = 81.490367; /*vapor Gas Const*/
}

real wetst_satP(real T)
{
  real psat;
  real a0 = 1.49560E+07;
  real a1 = -3.64411E+05;
  real a2 = 3.61522E+03;
  real a3 = -1.86330E+01;
  real a4 = 5.25909E-02;
  real a5 = -7.78881E-05;
  real a6 = 4.94851E-08;
  if (T > ws_Tc) T = ws_Tc ;
  psat=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
  return psat; /*Pa */
}
real wetst_satT(real P, real T)
{
real tsat;
real dT,dTM,dTA,dP,p1,p2,dPdT;
real dt = 1.e-4;
int i;
for (i=0; i<25; ++i)
{
if (T > ws_Tc) T = ws_Tc-0.5;
p1= wetst_satP(T) ;
p2= wetst_satP(T+dt) ;
dPdT = (p2-p1)/dt;
dP = P - p1 ;
dT = dP/dPdT;

dTA = fabs(dT);
dTM=0.1*T;
if(dTA>dTM) dT=dT*dTM/dTA;
T = T + dT;
if (fabs(dT)<TEMP_eps*T) break;
}
tsat = T;
return tsat; /*K */
}

real wetst_eosP(real rho, real T)
{
real P;
P = rho* Rgas_v * T ;
return P; /*Pa */
}

real wetst_eosRHO(real P, real T)
{
real rho;
rho = P/(Rgas_v * T);
return rho; /*kg/m3 */
}

real wetst_cpv(real T, real rho)
{
real cp;
real a0= 3.37519E+03;
real a1=-1.05456E+01;
real a2= 1.27550E-02;
cp = a0 + a1*T + a2*pow(T,2);
return cp; /* (J/Kg/K) */
}

real wetst_cvv(real T, real rho)
{
real cv;
cv = wetst_cpv(T,rho)-Rgas_v;
return cv; /* (J/Kg/K) */
}

real wetst_hv(real T,real rho)
{
real h;
h=T*wetst_cpv(T,rho);
return h; /* (J/Kg) */
}

real wetst_sv(real T, real rho)
{
real s;
real TDatum= 273.15;
real PDatum= 292800.0;
Appendix B. User Defined Functions for R134a

s=cpg*log(T/TDatum)+Rgas_v*log(PDatum/(Rgas_v*T*rho));
return s; /* (J/Kg/K) */
}

real wetst_muv(real T, real rho)
{
real muv;
muv=0.000011116;
return muv; /* (Kg/m/s) */
}

real wetst_ktv(real T, real rho)
{
real ktv;
ktv=0.012693;
return ktv; /* W/m/K */
}

real wetst_rhol(real T)
{
real rhol;
real a0=-2.19175E+04;
real a1=5.90411E+02;
real a2=-6.04919E+00;
real a3=3.25492E-02;
real a4=-9.74695E-05;
real a5=1.54095E-07;
real a6=-1.00631E-10;
if (T > ws_Tc) T = ws_Tc;
rhol=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
return rhol; /* (Kg/m3) */
}

real wetst_cpl(real T)
Appendix B. User Defined Functions for R134a

\[
\begin{align*}
\{ \\
\text{real } cpl; \\
\text{real } a0 &= 1.09375E+06; \\
\text{real } a1 &= -2.68356E+04; \\
\text{real } a2 &= 2.71286E+02; \\
\text{real } a3 &= -1.44502E+00; \\
\text{real } a4 &= 4.27829E-03; \\
\text{real } a5 &= -6.67762E-06; \\
\text{real } a6 &= 4.29438E-09; \\
\text{if } (T > \text{ws}_Tc) T &= \text{ws}_Tc; \\
cpl &= a0 + a1T + a2T^2 + a3T^3 + a4T^4 + a5T^5 + a6T^6; \\
\text{return } cpl; /* (J/Kg/K) */
\}
\]

\[
\begin{align*}
\text{real } \text{wetst_mul}(\text{real } T) \\
\{ \\
\text{real } mul &= 0.0; \\
\text{real } a0 &= 3.35945E+02; \\
\text{real } a1 &= -7.13719E+00; \\
\text{real } a2 &= 6.32769E+02; \\
\text{real } a3 &= -2.98423E+04; \\
\text{real } a4 &= 7.88006E-07; \\
\text{real } a5 &= -1.10341E-09; \\
\text{real } a6 &= 6.39691E-13; \\
\text{if } (T > \text{ws}_Tc) T &= \text{ws}_Tc; \\
\text{mul} &= a0 + a1T + a2T^2 + a3T^3 + a4T^4 + a5T^5 + a6T^6; \\
\text{return } mul; /* (Kg/m/s) */
\}
\]

\[
\begin{align*}
\text{real } \text{wetst_ktl}(\text{real } T) \\
\{ \\
\text{real } ktl; \\
\text{real } a0 &= 3.29632E-01; \\
\text{real } a1 &= -1.75906E-03;
\end{align*}
\]
real a2 = 5.44209E-06;
real a3 = -9.77113E-09;
real a4 = 6.46973E-12;
if (T > ws_Tc) T = ws_Tc;
ktl = a0 + a1 * T + a2 * pow(T, 2) + a3 * pow(T, 3) + a4 * pow(T, 4);
return ktl; /* W/m/K */
}

real wetst_surft(real T)
{
real sigma;
real a0 = 7.05060E-02;
real a1 = -3.80307E-04;
real a2 = 1.20556E-06;
real a3 = -3.16555E-09;
real a4 = 3.50343E-12;
if (T > ws_Tc) T = ws_Tc;
sigma = a0 + a1 * T + a2 * pow(T, 2) + a3 * pow(T, 3) + a4 * pow(T, 4);
return sigma; /* N/m */
}

/* do not change the order of the function list */

UDF_EXPORT WS_Functions WetSteamFunctionList =
{
wetst_init, /* initialization function */
wetst_satP, /* Saturation pressure */
wetst_satT, /* Saturation temperature */
wetst_eosP, /* equation of state */
wetst_eosRHO, /* equation of state */
wetst_hv, /* vapor enthalpy */
wetst_sv, /* vapor entropy */
wetst_cpv, /* vapor isobaric specific heat */
wetst_cvv, /* vapor isochoric specific heat */
Appendix B. User Defined Functions for R134a

B.1.2 Ideal Gas EOS - Real Gas Properties

/***********************************************************************/
/* User Defined R134a Properties: */
EOS: PERFECT GAS
PROPERTIES: REAL GAS (REFPROP)
Author: Marina Gallach Palma
Date: 10 July 2015*/
/***********************************************************************/

#include "udf.h"
#include "stdio.h"
#include "ctype.h"
#include "stdarg.h"

/*Global Constants for this model*/

real ws_TPP=169.85;
real ws_aaa=0.01;
real cpg=585.66;
/* Cp-vapor at low-pressure region*/

DEFINE_ON_DEMAND(I_do_nothing)
{ /* This is a dummy function to allow us to use */
 /* the Compiled UDFs utility*/

Appendix B. *User Defined Functions for R134a*

void wetst_init(Domain *domain)
{
    /* You must initialize these material property constants. They will be used in the wet steam model in Fluent */
    ws_Tc = 374.21; /*Critical Temp.*/
    ws_Pc = 4059300.0; /*Critical Pressure*/
    mw_f = 102.03; /*Fluid droplet molecular weight (water)*/
    Rgas_v = 81.490367; /*Vapor Gas Const*/
}

real wetst_satP(real T)
{
    real psat;
    real a0 = 1.49560E+07;
    real a1 = -3.64411E+05;
    real a2 = 3.61522E+03;
    real a3 = -1.86330E+01;
    real a4 = 5.25909E-02;
    real a5 = -7.78881E-05;
    real a6 = 4.94851E-08;
    if (T > ws_Tc) T = ws_Tc;
    psat = a0 + a1*T + a2*pow(T,2.0) + a3*pow(T,3.0) + a4*pow(T,4.0) + a5*pow(T,5.0) + a6*pow(T,6.0);
    return psat; /*Pa*/
}

real wetst_satT(real P, real T)
{
    real tsat;
    real dT, dTM, dTA, dP, p1, p2, dPdT;
    real dt = 1.0e-4;
    int i;
    for (i=0; i<25; ++i)
{  
if (T > ws_Tc) T = ws_Tc-0.5;
  
p1= wetst_satP(T) ;
  
p2= wetst_satP(T+dt) ;
  
dPdT = (p2-p1)/dt;
  
dP = P - p1 ;
  
dT = dP/dPdT;
  
dTA = fabs(dT);
  
dTM=0.1*T;
  
if(dTA>dTM) dT=dT*dTM/dTA;
  
T = T + dT;
  
if (fabs(dT)<TEMP_eps*T) break;
}

  
tsat = T;
  
return tsat; /*K */
  
}

real wetst_eosP(real rho, real T) 
{
real P;
  
P = rho* Rgas_v * T ;
  
return P; /*Pa */
  
}

real wetst_eosRHO(real P, real T) 
{
real rho;
  
rho = P/(Rgas_v * T) ;
  
return rho; /*kg/m3 */
  
}

real wetst_cpv(real T, real rho) 
{

}
Appendix B. User Defined Functions for R134a

real cp;
real act=-4.42590E+02;
real at= 7.36158E+00;
real at2=-8.68716E-03;
real ad= -6.22141E+00;
real ad2= 6.98903E-02;
if (T > 455.0) T = 455.0;
cp = act + at*T + at2*pow(T,2.0) + ad*rho + ad2*pow(rho,2.0);
return cp; /* (J/Kg/K) */
}

real wetst_cvv(real T, real rho)
{
real cv;
real act= 2.14813E+02;
real at= 1.90981E+00;
real at2=-3.42800E-04;
real ad= 5.02810E+00;
real ad2=-5.52774E-04;
real atd=-1.06197E-02;
if (T > 455.0) T = 455.0;
cv = act + at*T + at2*pow(T,2.0) + ad*rho + ad2*pow(rho,2.0) + atd*T*rho;
return cv; /* (J/Kg/K) */
}

real wetst_hv(real T,real rho)
{
real h;
real act= 1.51022E+05;
real at= 9.28061E+02;
real ad= -2.74230E+02;
if (T > 455.0) T = 455.0;
h = act + at*T + ad*rho;
return h; /* (J/Kg) */
real wetst_sv(real T, real rho)
{
    real s ;
    real act = 1.82994E+03;
    real at = -1.07844E+00;
    real at2 = 4.62066E-03;
    real ad = -6.35613E+00;
    real ad2 = 1.30475E-02;
    real atd = 5.10279E-03;
    if (T > 455.0) T = 455.0;
    s = act + at*T + at2*pow(T,2.0) + ad*rho + ad2*pow(rho,2.0) + atd*T*rho;
    return s; /* (J/Kg/K) */
}

real wetst_muv(real T, real rho)
{
    real muv;
    real act = -9.62866E-08;
    real at = 4.13590E-08;
    real at2 = -4.63950E-12;
    real ad = -2.85153E-08;
    real ad2 = 7.33865E-11;
    real atd = 8.16447E-11;
    if (T > 455.0) T = 455.0;
    muv = act + at*T + at2*pow(T,2) + ad*rho + ad2*pow(rho,2) + atd*T*rho;
    return muv; /* (Kg/m/s) */
}

real wetst_ktv(real T, real rho)
{
    real ktv;
    real act = -9.30258E-03;
real at = 7.09679E-05;
real at2 = 1.65925E-08;
real ad = 5.59051E-05;
real ad2 = 2.42633E-07;
real atd = -1.66511E-07;
if (T > 455.0) T = 455.0;
ktv = act + at * T + at2 * pow(T, 2) + ad * rho + ad2 * pow(rho, 2) + atd * T * rho;
return ktv; /* W/m/K */
}

real wetst_rhol(real T)
{
real rhol;
real a0 = -2.19175E+04;
real a1 = 5.90411E+02;
real a2 = -6.04919E+00;
real a3 = 3.25492E-02;
real a4 = -9.74695E-05;
real a5 = 1.54095E-07;
real a6 = -1.00631E-10;
if (T > ws_Tc) T = ws_Tc;
rhol = a0 + a1 * T + a2 * pow(T, 2) + a3 * pow(T, 3) + a4 * pow(T, 4) + a5 * pow(T, 5) + a6 * pow(T, 6);
return rhol; /* (Kg/m3) */
}

real wetst_cpl(real T)
{
real cpl;
real a0 = 1.09375E+06;
real a1 = -2.68356E+04;
real a2 = 2.71286E+02;
real a3 = -1.44502E+00;
real a4 = 4.27829E-03;
real a5 = -6.67762E-06;
real a6= 4.29438E-09;
if (T > ws_Tc) T = ws_Tc ;
cpl=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
return cpl; /* (J/Kg/K) */
}

real wetst_mul(real T)
{
real mul=0.0;
real a0= 3.35945E+02;
real a1=-7.13719E+00;
real a2= 6.32769E-02;
real a3=-2.98423E-04;
real a4= 7.88006E-07;
real a5=-1.10341E-09;
real a6= 6.39691E-13;
if (T > ws_Tc) T = ws_Tc ;
mul=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
return mul; /* (Kg/m/s) */
}

real wetst_ktl(real T)
{
real ktl;
real a0= 3.29632E-01;
real a1=-1.75906E-03;
real a2= 5.44209E-06;
real a3=-9.77113E-09;
real a4= 6.46973E-12;
if (T > ws_Tc) T = ws_Tc ;
ktl = a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4); return ktl; /* W/m/K */
}
real wetst_surft(real T)
{
    real sigma;
    real a0= 7.05060E-02;
    real a1=-3.80307E-04;
    real a2= 1.20556E-06;
    real a3=-3.16555E-09;
    real a4= 3.50343E-12;
    if (T > ws_Tc) T = ws_Tc;
    sigma=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4);
    return sigma; /* N/m */
}

/* do not change the order of the function list */

UDF_EXPORT WS_Functions WetSteamFunctionList =
{
    wetst_init, /* initialization function */
    wetst_satP, /* Saturation pressure */
    wetst_satT, /* Saturation temperature */
    wetst_eosP, /* equation of state */
    wetst_eosRHO, /* equation of state */
    wetst_hv, /* vapor enthalpy */
    wetst_sv, /* vapor entropy */
    wetst_cpv, /* vapor isobaric specific heat */
    wetst_cvv, /* vapor isochoric specific heat */
    wetst_muv, /* vapor dynamic viscosity */
    wetst_ktv, /* vapor thermal conductivity */
    wetst_rhol, /* sat. liquid density */
    wetst_cpl, /* sat. liquid specific heat */
    wetst_mul, /* sat. liquid viscosity */
    wetst_ktl, /* sat. liquid thermal conductivity */
    wetst_surft /* liquid surface tension */
};
B.2 UDF for real gas

/**********************************************************************/
/*
User Defined R134a Properties:
EOS: PENG ROBINSON
PROPERTIES: REAL GAS
Author: Marina Gallach Palma
Date : 11 July 2015*/
/**********************************************************************/

#include "udf.h"
#include "stdio.h"
#include "ctype.h"
#include "stdarg.h"

/*Global Constants for this model*/

real ws_TPP=169.85;
real ws_aaa=0.01;
real cpg=585.66;
real rho_min=1;
real abs_tol=1e-5;

/* Cp-vapor at low-pressure region*/

DEFINE_ON_DEMAND(I_do_nothing)
{ /* This is a dummy function to allow us to use */
 /* the Compiled UDFs utility*/
 }

void wetst_init(Domain *domain)
{
 /* You must initialize these material property constants..
 they will be used in the wet steam model in fluent */
Appendix B. User Defined Functions for R134a

ws_Tc = 374.21; /*Critical Temp. */
ws_Pc = 4059300.0; /*Critical Pressure */
mw_f = 102.03; /*fluid droplet molecular weight (water) */
Rgas_v = 81.490367; /*vapor Gas Const*/
}

real wetst_satP(real T)
{
    real psat;
    real a0 = 1.49560E+07;
    real a1 =-3.64411E+05;
    real a2 = 3.61522E+03;
    real a3 =-1.86330E+01;
    real a4 = 5.25909E-02;
    real a5 =-7.78881E-05;
    real a6 = 4.94851E-08;
    if (T > ws_Tc) T = ws_Tc ;
    psat=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
    return psat; /*Pa */
}

real wetst_satT(real P, real T)
{
    real tsat;
    real dT,dTM,dTA,dP,p1,p2,dPdT;
    real dt = 1.e-4;
    int i ;
    for (i=0; i<25; ++i)
    {
        if (T > ws_Tc) T = ws_Tc-0.5;
        p1= wetst_satP(T) ;
        p2= wetst_satP(T+dt) ;
        dPdT = (p2-p1)/dt;
        dP = P - p1 ;
    }
dT = dP/dPdT;

dTA = fabs(dT);
dTM=0.1*T;
if(dTA>dTM) dT=dT*dTM/dTA;
T = T + dT;
if (fabs(dT)<TEMP_eps*T) break;
}
tsat = T;
return tsat; /*K */
}

real wetst_eosP(real rho, real T)
{
real P;
/*P = rho* Rgas_v * T ;*/
real a = (0.457235*Rgas_v*Rgas_v*ws_Tc*ws_Tc)/ws_Pc;
real b = (0.077796*Rgas_v*ws_Tc)/ws_Pc;
real acentric = 0.32684;
real k = 0.37464+1.54226*acentric-0.26992*acentric*acentric;
real alpha = (1.0+k*(1.0-sqrt(T/ws_Tc)))*(1.0+k*(1.0-sqrt(T/ws_Tc)));
if(rho<rho_min) rho=rho_min;
real v=1.0/rho;
P = ((Rgas_v*T)/(v-b))-(a*alpha/(v*v+2.0*v*b-b*b));
return P; /*Pa */
}

real wetst_eosRHO(real P, real T)
{
real v=1.0/rho_min;
real a = (0.457235*Rgas_v*Rgas_v*ws_Tc*ws_Tc)/ws_Pc;
real b = (0.077796*Rgas_v*ws_Tc)/ws_Pc;
real acentric = 0.32684;
real k = 0.37464+1.54226*acentric-0.26992*acentric*acentric;
real alpha = (1.0+k*(1.0-sqrt(T/ws_Tc)));
real a_2= b - Rgas_v*T/P;
real a_1=-3.0*b*b - 2.0*Rgas_v*b*T/P + alpha*a/P;
real a_0= b*b*b + Rgas_v*T*b*b/P - alpha*a*b/P;
real temp=0.0;
while(fabs(v-temp)>abs_tol)
{
    real polynomial = v*v*v + a_2*v*v + a_1*v + a_0;
    real derivative = 3.0*v*v + 2.0*a_2*v + a_1;
    temp = v;
    v = temp - polynomial/derivative;
}
real rho=1.0/v;
return rho; /*kg/m3 */
}

real wetst_cpv(real T, real rho)
{
real cp;
real act=-4.42590E+02;
real at= 7.36158E+00;
real at2=-8.68716E-03;
real ad= -6.22141E+00;
real ad2= 6.98903E-02;
    cp = act + at*T + at2*pow(T,2) + ad*rho + ad2*pow(rho,2);
    return cp; /* (J/Kg/K) */
}

real wetst_cvv(real T, real rho)
{
real cv;
real act= 2.14813E+02;
real at= 1.90981E+00;
real at2=-3.42800E-04;
real ad = 5.02810E+00;
real ad2 = -5.52774E-04;
real atd = -1.06197E-02;
cv = act + at*T + at2*pow(T, 2) + ad*rho + ad2*pow(rho, 2) + atd*T*rho;
return cv; /* (J/Kg/K) */
}

real wetst_hv(real T, real rho)
{
real h;
real act = 1.51022E+05;
real at = 9.28061E+02;
real ad = -2.74230E+02;
h = act + at*T + ad*rho;
return h; /* (J/Kg) */
}

real wetst_sv(real T, real rho)
{
real s;
real act = 1.82994E+03;
real at = -1.07844E+00;
real at2 = 4.62066E-03;
real ad = -6.35613E+00;
real ad2 = 1.30475E-02;
real atd = 5.10279E-03;
s = act + at*T + at2*pow(T, 2) + ad*rho + ad2*pow(rho, 2) + atd*T*rho;
return s; /* (J/Kg/K) */
}

real wetst_muv(real T, real rho)
{
real muv;
real act = -9.62866E-08;
real at= 4.13590E-08;
real at2=-4.63950E-12;
real ad= -2.85153E-08;
real ad2= 7.33865E-11;
real atd= 8.16447E-11;
muv = act + at*T + at2*pow(T,2.) + ad*rho + ad2*pow(rho,2.) + atd*T*rho;
return muv; /* (Kg/m/s) */
}

real wetst_ktv(real T, real rho)
{
real ktv;
real act=-9.30258E-03;
real at= 7.09679E-05;
real at2= 1.65925E-08;
real ad= 5.59051E-05;
real ad2= 2.42633E-07;
real atd=-1.66511E-07;
ktv=act+at*T+at2*pow(T,2.)+ad*rho+ad2*pow(rho,2.)+atd*T*rho;
return ktv; /* W/m/K */
}

real wetst_rhol(real T)
{
real rhol;
real a0=-2.19175E+04;
real a1= 5.90411E+02;
real a2=-6.04919E+00;
real a3= 3.25492E-02;
real a4=-9.74695E-05;
real a5= 1.54095E-07;
real a6=-1.00631E-10;
if (T > ws_Tc) T = ws_Tc;
rhol=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
return rho1; /* (Kg/m3) */
}

real wetst_cpl(real T)
{
    real cpl;
    real a0= 1.09375E+06;
    real a1=-2.68356E+04;
    real a2= 2.71286E+02;
    real a3=-1.44502E+00;
    real a4= 4.27829E-03;
    real a5=-6.77762E-06;
    real a6= 4.29438E-09;
    if (T > ws_Tc) T = ws_Tc ;
    cpl=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
    return cpl; /* (J/Kg/K) */
}

real wetst_mul(real T)
{
    real mul=0.0;
    real a0= 3.35945E+02;
    real a1=-7.13719E+00;
    real a2= 6.32769E-02;
    real a3=-2.98423E-04;
    real a4= 7.88006E-07;
    real a5=-1.10341E-09;
    real a6= 6.39691E-13;
    if (T > ws_Tc) T = ws_Tc ;
    mul=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4)+a5*pow(T,5)+a6*pow(T,6);
    return mul; /* (Kg/m/s) */
}

real wetst_ktl(real T)
Appendix B. User Defined Functions for R134a

{  
real ktl;
real a0= 3.29632E-01;
real a1=-1.75906E-03;
real a2= 5.44209E-06;
real a3=-9.77113E-09;
real a4= 6.46973E-12;
if (T > ws_Tc) T = ws_Tc ;
ktl = a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4);
return ktl; /* W/m/K */
}

real wetst_surft(real T)
{
real sigma;
real a0= 7.05060E-02;
real a1=-3.80307E-04;
real a2= 1.20556E-06;
real a3=-3.16555E-09;
real a4= 3.50343E-12;
if (T > ws_Tc) T = ws_Tc;
sigma=a0+a1*T+a2*pow(T,2)+a3*pow(T,3)+a4*pow(T,4);
return sigma ;/* N/m */
}

/* do not change the order of the function list */

UDF_EXPORT WS_Functions WetSteamFunctionList =
{
wetst_init, /*initialization function*/
wetst_satP, /*Saturation pressure*/
wetst_satT, /*Saturation temperature*/
wetst_eosP, /*equation of state*/
wetst_eosRHO, /*equation of state*/
}
wetst_hv, /*vapor enthalpy*/
wetst_sv, /*vapor entropy*/
wetst_cpv, /*vapor isobaric specific heat*/
wetst_cvv, /*vapor isochoric specific heat*/
wetst_muv, /*vapor dynamic viscosity*/
wetst_ktv, /*vapor thermal conductivity*/
wetst_rhol, /*sat. liquid density*/
wetst_cpl, /*sat. liquid specific heat*/
wetst_mul, /*sat. liquid viscosity*/
wetst_ktl, /*sat. liquid thermal conductivity*/
wetst_surft /*liquid surface tension*/
};
Bibliography


