# An unstructured tabulation method for the computation of thermo-physical fluid properties

Aryaman Madan

Master of Science Thesis

TU Delft Delft University of Technology

Challenge the future

# An unstructured tabulation method for the computation of thermo-physical fluid properties

by

Aryaman Madan

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday June 29, 2018 at 14:30 PM.

Student number: 4422007

| | | |
|---|---|---|
| Thesis committee: | Dr. Matteo Pini, | Supervisor |
| | Ir. Antonio Rubino, | Supervisor |
| | Prof. Dr. Piero Colonna, | Head of Examination Committe |
| | Dr. Rene Pecnik | Reader |



# Abstract

In various propulsion and power systems, modeling of non-ideal fluid flows (fluids that depart from ideal gas behaviour), presents a great challenge. For example, in organic rankine cycle (ORC) turbines, where a part of the expansion process occurs close to the vapour saturation curve, the flow devietes highly from ideal behavior. A branch of fluid dynamics called the Non-ideal compressible fluid dynamics (NICFD) deals with the modeling and analysis of such non ideal fluid flows.

As a consequence of the need for accurate thermo-physical property computation, various models have recently been developed for non ideal flows and a number of libraries are available to accurately predict the thermo-physical properties. However, the available thermodynamic libraries are often computationally costly since they require solving of complex equations of state (EoS) to obtain thermo-physical properties. When these libraries are coupled with existing simulation codes, (for example in computational fluid-dynamics), the simulation process is computationally inefficient. This thesis is an endeavor towards enhancing the computational efficiency of the process of thermodynamic property calculation with the use of the Look up table (LUT) approach.

The LUT method aims at computing thermodynamic properties of a fluid with the help of array indexing operations applied on pre calculated or existing thermodynamic tables. These tables are initially obtained from a thermodynamic library (*FluidProp*). A binary search algoriths helps in accurately locating the query point(s) on the thermodynamic domain. A data interpolation algorithm is then used to predict the thermodynamic properties of interest. The presented LUT method ensures inherently high accuracy with a very small computational cost, as demonstrated later in the thesis.

To check the applicability of the LUT method, it is used to obtain the pressure variation across a control volume with subsonic flow conditions. As a second and a much larger application, the LUT tool is coupled with an in house MOC (Method of Characteristics) tool to desin the geometry of a supersonic nozzle.

A comprehensive analysis of this method is presented by comparing the accuracies and computational cost with the results from *FluidProp*. Both interpolation methods implemented in the proposed LUT method prove to be computationally efficient and accurate. The method is successfully applied to the MOC tool to design the geometry of the diverging section of a supersonic nozzle.

# Acknowledgements

I would like to express my deepest graduate to Dr. Matteo Pini to have given me this great opportunity to work under his guidance and support. I would like to thank my PhD supervisor Ir. Antonio Rubino for his continuous guidance throughout the duration of the thesis. His regular, critical and honest feedback made this work possible. During the really difficult times, his support and encouragement was one of the main driving forces to reach for the light at the end of the tunnel.

I would like to extend my sincere thanks to Mr. John Stalls, the student counsellor at the TU Delft central international office. His advice and support amidst all successes and failures have been of prime importance during the last two years.

I am grateful to my examination committee, Prof. Piero Colonna and Dr. Rene Pecnik for devoting their precious time for the consideration and evaluation of this work.

Special thanks to my friend Pranav Balvalli in Delft for his constant support, enthusiasm and help during this period. Finally, I would like to thank my beloved family for their unconditional love, care and support during this important and defining phase of life.

# Contents

# List of Figures

# List of Tables

" I dedicate this thesis to my parents, brother and grandparents, for always being there for me"

— *Aryaman*

# Nomenclature

## List of Acronyms

| | |
|---|---|
| **NICFD** | Non Ideal Compressible Fluid Dynamics |
| **CFD** | Computational Fluid Dynamics |
| **ORC** | Organic Rankine Cycle |
| **MDM** | Octamethyltrisiloxane |
| **EoS** | Equations of State |
| **NN** | Nearest Neighbour |
| **MEOS** | Multi-parameter Equation of State |
| **SU2** | Stanford University Unstructured |
| **FP** | *FluidProp* |
| **MOC** | Method of Characteristics |
| **SW-EoS** | Span-Wagner Equation of State |
| **NIST** | National Institute of Standards and Technology |
| **NR** | Newton Raphson |
| **PDE** | Partial Differential Equation |
| **SBO** | Surrogate Based Optimization |
| **DoE** | Design of experiments |
| **IDW** | Inverse Data Weighing |
| **CO2** | Canbon Dioxide |
| **MM** | Hexamethyldisiloxane |
| **D6** | Octamethylcyclotetrasiloxane |
| **BC-NC** | Bicubic Non-Consistent |
| **MRE** | Mean Relative Error |

# List of Symbols

| | | |
|---|---|---|
| $p$ | Presure | [Pa] |
| $v$ | Specific volume | [$m^3$/kg] |
| $T$ | Temperature | [K] |
| $R_0$ | Universal gas constant | [J/kg.K] |
| $R$ | Gas constant | [J/kg.mol] |
| $MM$ | Molar mass | [kg/mol] |
| $KE$ | Kinetic Energy | [J] |
| $z$ | Compressibility factor | [-] |
| $s$ | Specific entropy | [J/kg.K] |
| $\rho$ or $d$ | Density | [kg/$m^3$] |
| $\delta$ | Reduced density | [-] |
| $p_t$ | Total pressure | [Pa] |
| $p_s$ | Static pressure | [Pa] |
| $\dot{m}$ | Mass flow rate | [kg/s] |
| $A$ | Area | [$m^2$] |
| $M$ | Mach number | [-] |
| $\gamma$ | Ratio of specic heats | [-] |
| $u$ | velocity | [m/s] |
| $h$ | Specific static enthalpy | [J/kg] |
| $h_0$ | Total enthalpy | [J/kg] |
| $c_p$ | Specific heat capacity at constant pressure | [J/kg.K] |
| $c_v$ | Specific heat capacity at constant volume | [J/kg.K] |
| $M_{design}$ | Design mach number | [-] |
| $P_r$ | Reduced pressure | [-] |
| $T_r$ | Reduced temperature | [-] |
| $A_{th}$ | Throat area | [$m^2$] |
| $A_e$ | Exit area | [$m^2$] |
| $A_{in}$ | Inlet area | [$m^2$] |

# 1 Introduction

## 1.1. Background

*Non-ideal compressible fluid dynamics* (NICFD) is the branch of fluid dynamics that deals with the flow characteristics of fluid flows that deviate from the ideal fluid conditions in the compressible flow regime. NICFD is particularly important for the study of the actual characteristics of dense vapours, supercritical flows and compressible two-phase flows, whereby the thermodynamic behaviour of the fluid differs considerably from that of a perfect gas [8] . To correctly predict the thermodynamic behaviour of the fluids deviating from flow ideality, a substantial understanding of non ideal fluid thermodynamics is necessary [16] [15] .

At low pressures and high temperatures, fluids exhibit ideal gas behavior as the compressibility factor $z = 1$. Under such thermodynamic conditions, the fluid follows the ideal gas law and the flow can be predicted by the ideal gas equation of state (EoS). However, under the thermodynamic state where, where $z \neq 1$, the flow departs from the ideal gas law. For example, in Organic Rankine Cycle turbines (ORC), a part of the expansion process occurs typically in the proximity of the vapour saturation curve or the critical point. In these thermodynamic conditions, the fluid behavior is known to significantly depart from the ideal gas model. The Equations of State (EoS) and their derivatives are most commonly used to accurately predict the thermodynamic properties for non-ideal fluid flows. However, the use of these EoS through thermodynamic libraries like *FluidProp* [5] is computationally costly for detailed numerical studies, for instance in Computational Fluid Dynamics(CFD) analysis.

To reduce the computational cost, the Look Up Tables(LUT) can be used as an alternate approach to determine the thermodynamic properties. The LUT method has a simple application and it is capable of calculating the thermo-physical properties accurately.

## 1.2. Look Up Tables Overview

For applications involving complex thermodynamics, the computational cost for the calculation of thermo-physical properties is often high. The thermo-physical properties for real gases can be calculated by with the help of EoS and as their respective derivative forms. However the direct solution through EoS is often complex and computationally costly [15]. To increase the computational efficiency of thermo-physical property calculation, an alternative approach in the form of Look-up Tables (LUT) can be used to predict the fluid behaviour [15] [7].

A thermodynamically consistent LUT method to compute thermo-physical properties is proposed by Pini et. al. [15]. For this method, a thermodynamic mesh is generated on the $log(v) - s$ domain. The values of specific internal energy $e$, which is expressed in the form of specific entropy $s$ and specific volume $v$ is obtained for each node with the help of the EoS implemented in the thermodynamic library *FluidProp* [5]. Following this, the values at the nodes are used to formulate a fundamental relation for $e = e(v, s)$. This fundamental relation is obtained for each cell of the grid. The bi-cubic bivariate interpolation form for the fundamental relation is used to obtain first derivatives, which are then used to compute the rest of the thermodynamic properties. Similarly, to obtain the transport properties, pini et. al. propose the formulation of a bilinear

functional forms of dynamic viscosity $\mu$ and the thermal conductivity $k$. The reader is referred to [15] for more information on these formulations.

This LUT method uses the k-d tree algorithm [3] [13] for locating the query vector on the thermodynamic grid. As explained by the author, the thermodynamic pair $(v, s)$ is necessary for the formulation of the fundamental relation $e(v, s)$. If the input pair is not of the form $(v, s)$, the input pair is converted to the form $(v, s)$ through iterative solving of a non-linear system of equations.

The major advantages with the implementation of the LUT approach proposed by Pini et. al. [15] is that the method is able to maintain thermodynamic consistency and stability. The author demonstrates the model's applicability through its use in blade to blade calculations and through flow calculations in multi stage machines. Since $(log(v), s)$ is selected as the domain on which the thermodynamic mesh is generated and the fundamental relation $e(v, s)$ is built using $v, s$ as independent variables, this method requires the input pair to be in the form of $(v, s)$. An input pair of any other couple needs to be converted to the form $(v, s)$ by solving a system of non-linear equations, which can add to the computational time by a small margin. However, overall this LUT approach is computationally efficient and maintains a good accuracy in thermo-physical property calculations.

Miyagawa and Hill [12] present an LUT method to compute the thermodynamic properties of steam with with the help of the Taylor's series expansion. On a structured grid, for a given interval of temperature and pressure in the superheated region, this method helps in calculating the steam properties accurately with a small computational cost. The initial values of the properties (and derivatives) to generate thermodynamic tables are obtained using the Helmholtz energy formulations. A structured grid is built on the $p, T$ domain with pressure and temperature as independent variables. The centre of each cell contains the nodal point which stores the thermodynamic properties and their derivatives for each cell. A truncated expansion method for Taylor series is then used to compute the thermodynamic properties. The method works sufficiently well in superheated regions, however according to Miyagawa and Hill [12], the method with pressure as the independent variable is not well suited for computation of properties in the critical regions.

Miyagawa and Hill [20] presented another Look Up table method that makes use of the Taylor series expansion to compute the temperature, specific volume and entropy. For this method, pressure and enthalpy are used as the independent variables. Similar to the previous LUT method proposed by the author, a structured grid on $(p, h)$ domain is built for this method. The nodal point is considered to be in the middle of each cell where the initial thermodynamic properties are stored. A truncated Taylor series (upto 2 orders) is then used to evaluate the temperature, specific volume and the entropy for the region of interest [20]. The major drawback of the method is that for the calculation of the desired properties, it is also required to store the first and second derivative values initially on the nodes.

## 1.3. Motivation

As mentioned in section 1.1, the use of EoS through thermodynamic libraries renders the process of property calculation computationally costly. Processes such as CFD analysis, which require iterative solving of governing equations are often computationally expensive [15]. Therefore, it becomes important to use a different approach to predict thermodynamic properties in order to reduce computational time, while aiming to maintain an optimum accuracy with respect to the EoS solutions.

A common feature for the LUT methods mentioned in section 1.1 is the use of first and second order derivatives of the properties for obtaining the thermodynamic properties. Another highlight is the use of structured grids to mesh the domain and generate thermodynamic tables.

Proposed as an **original** part of this project is the application of the LUT method on an unstructured grid for thermo-physical property calculation. Secondly, one of the two interpolation schemes used in the proposed LUT method is a simple inverse data weighing interpolation scheme called Shephard's interpolation [17] . This method is fairly simple in its application and does not require any functional formulation or prestored derivative values for interpolation. As opposed to the complex process of solving the EoS to predict thermodynamic properties, the LUT method uses simple array indexing operations on the thermodynamic tables. This approach is fast as it uses a binary search algorithm (for example, k-d tree) and an interpolation scheme (like Shephard's Interpolation) on the data obtained from the thermodynamic tables. Upon successful implementation, the LUT method for thermo-physical property calculation can have many applications.

## 1.4. Scope and Objectives

The application of any LUT approach can be broken down into a series of steps listed in table 1.1:

| Step | Task |
|---|---|
| 1 | Mesh generation |
| 2 | Table generation or loading |
| 3 | Search algorithm application for nearest-neighbour search |
| 4 | Looking up data or using an interpolation method to calculate the required data |

Table 1.1: Steps involved in a Look Up Table approach for obtaining the required data of interest

To compute thermodynamic properties using the LUT method, a mesh is built on the thermodynamic domain of interest. Each node of the thermodynamic mesh stores the thermodynamic properties which are obtained from *FluidProp* as a part of pre-processing. Next, a suitable search algorithm is used to search for the point of interest (or the query point) on the thermodynamic domain and find its nearest neighbours. Finally, an interpolation method such as the Shephard's interpolation [17] [10] is applied to compute the set of thermodynamic properties of interest [11].

The overall LUT algorithm aims to predict the thermodynamic properties with low computational cost and an optimum accuracy with respect to *FluidProp*. For testing the applicability of the LUT method, this work also aims at integrating the LUT approach with an in house tool known as the MOC(Method of Characteristics) tool for designing supersonic nozzle geometries [1].

This thesis work is aimed at answering the following research questions:

1. How does the application of an unstructured tabulation method affect the computational cost for the thermo-physical property calculation?
2. How does the change in mesh density affect the accuracy of the LUT algorithm?
3. What is the applicability of the LUT approach for design problems? For example, can it be applied to a supersonic nozzle design tool like MOC to reduce the computational cost of the overall design?

## 1.5. Outline

**Chapter 2** of the thesis outlines the theoretical background on topics of flow ideality, interpolation methods and the binary search algorithm. **Chapter 3** describes the overall methodology of LUT algorithm. The chapter is divided into sub parts which describe the unstructured mesh generation, application of a search algorithm and interpolation methods on the LUT for thermodynamic property calculation. **Chapter 4** contains the analysis of the LUT method. Here, the accuracy of thermodynamic properties and the computational time of the LUT method are evaluated. **Chapter 5** presents two applications of the LUT tool and the associated results. First is an expansion problem which is self contained within the LUT tool. Second is the integration of LUT tool with MOC tool [1] to design the diverging section of a supersonic nozzle using Toluene as the working fluid under the given thermodynamic conditions. **Chapter 6** presents all the conclusions of the work done in the thesis. It also highlights the shortcomings or limitations of the work and then points out some recommendations for future work.

# 2

# Theoretical Background

*This chapter discusses the theory of to non-ideal fluids and the description of tools (search algorithm and interpolation methods) using which the LUT method is implemented in chapter 3.*

## 2.1. Ideal and Non-ideal fluids

### 2.1.1. Flow Ideality

It is observed from experimental observations that $p-v-T$ behaviour of gases at low pressures, closely follows the relation given by equation 2.1. A fluid which follows the equation 2.1 at all pressures and temperatures is called an *ideal fluid*.

$$Pv = RT \tag{2.1}$$

where $R$ is the gas constant and is different for each gas. It is given by equation 2.2 as follows:

$$R = \frac{R_0}{M} \tag{2.2}$$

Equation 2.1 is called the *ideal gas equation of state* and can be established using the postulates of the *kinetic theory of matter*, with two important assumptions that there are no intra-molecular cohesive forces and that the molecules have a negligible volume as compared to that of the gas. At very small pressure or very large temperature, the intermolecular forces of attraction and the molecular volumes are not of great significance and the gas closely obeys the ideal gas equation. However, when the pressure increases, the intermolecular forces of attraction and repulsion increase and the volume of the molecules becomes non negligible as compared to the volume of the gas. This leads to a considerable deviation from the real gas behaviour [4].

### 2.1.2. Non-ideal Flow

The amount of deviation of a real gas from the ideal gas is expressed in terms of the compressibility factor $z$. $z$ is given by equation [4][14]:

$$Pv = zRT \tag{2.3}$$

The compressibility factor can also be defined by equation 2.4.

$$z = \frac{v}{v_{ideal}} \tag{2.4}$$

where

$$v_{ideal} = \frac{RT}{p} \tag{2.5}$$

For an ideal gas, the value of $z = 1$. The value of $z$ can be can be above or below 1 for a non ideal gas, depending upon the deviation from ideal behaviour. The variation of $z$ with pressure, for a given temperature is given by the compressibility charts. Since different gases behave differently at different temperatures and pressures, the compressibility charts are different for different gases. But, if the pressure and temperature of the gases is normalized with respect to their critical pressure and temperature, $z$ is approximately the same.This is known

as the *Principle of Corresponding States.* These normalized pressure and temperature are given by equations 2.6 and 2.7, respectively [4][14].

$$P_r = \frac{P}{P_c} \tag{2.6}$$

$$T_r = \frac{T}{T_c} \tag{2.7}$$

$P_c$ and $T_c$ are the critical pressure and the critical temperature of the gas respectively.
It is noted from the compressibility charts that fluids have maximum deviation from ideal behaviour in the vicinity of the critical point and the saturation curves.

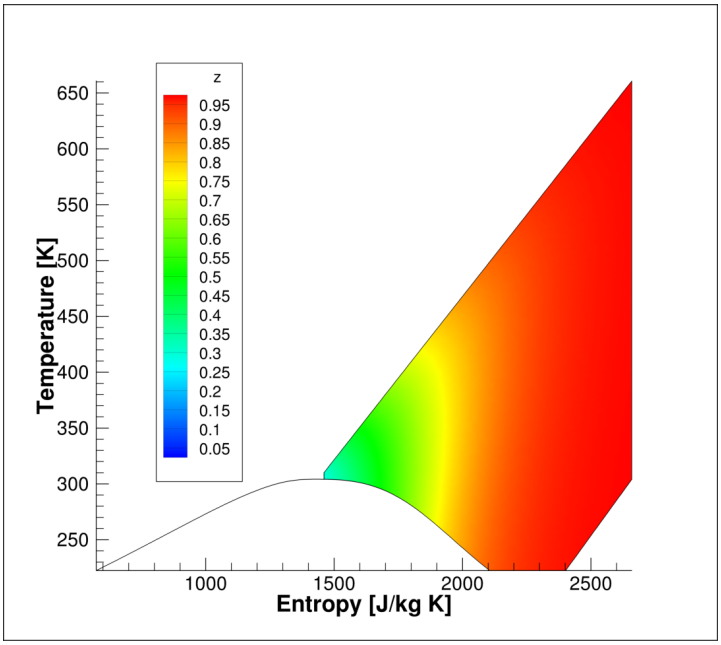



Figure 2.1: Compressibility factor for Carbon Dioxide

Figure 2.1 shows the compressibility factor for carbon dioxide on the $T-s$ thermodynamic domain. It is seen that in the region far from critical point, $z = 1$ and hence the fluid can be modelled in this region using the Ideal gas EoS. However, close to the critical point, $z$ shows maximum deviation from unity and therefore departs from the ideal gas behaviour. It is essential to use appropriate EoS for accurately predicting thermodynamic properties for such regions where the fluid departs significantly from ideal gas law.

### van der Walls EoS

The van der Walls equation is one of the most common EoS which are applicable to real gases. It is basically an improved version of equation 2.1 [14]. By taking into account the *intermolecular forces of attraction* and *the volume occupied by the molecules*, the van der Walls equation [14] [4] is given as follows:

$$(P + \frac{a}{v^2})(v - b) = RT \tag{2.8}$$

The term $\frac{a}{v^2}$ accounts for the inter-molecular forces and the term b accounts for the volume occupied by the molecules. The values of these parameters can be empirically obtained for different gases. At the critical point however, the values are determined by eliminating the volume term by calculating the first and the second derivatives of the pressure with respect to the volume. This yields the following values of a and b as represented by equation 2.9 .

$$\begin{aligned} a &= \frac{27R^2T_{cr}^2}{64P_{cr}^2} \\ b &= \frac{RT_{cr}}{8P_{cr}} \end{aligned} \tag{2.9}$$

#### Span Wagner Equation of State (SW)

Span-Wagner EoS is described as a fundamental equation in the form of Helmholtz energy, $\alpha$, in non dimensional form [18] . The reduced Helmholtz energy is split into two parts. The first part predicts the behaviour of a hypothetical ideal gas (superscript o) for a given temperature and density and the second part predicts the residual behaviour (superscript r) of the non-ideal fluid [18]. Therefore, the EoS is written in the form:

$$\frac{a(T,\rho)}{RT} = \frac{\alpha^0(T,\rho) + \alpha^r(T,\rho)}{RT} = a^0(\tau,\delta) + a^r(\tau,\delta) \tag{2.10}$$

where $a$ is the specific or molar Helmholtz energy, $R$ is the corresponding gas constant, $T$ is the temperature, $\rho$ is the density, $\tau$ is the inverse reduced temperature, and $\delta$ the reduced density. Equations 2.11 -2.12 represent the inverse reduced temperature and reduced density respectively.

$$\tau = \frac{T_r}{T} \tag{2.11}$$

$$\delta = \frac{\rho}{\rho_r} \tag{2.12}$$

Using Helmholtz energy as a function of temperature and density is suitable for the formulation of the fundamental equations [6]. With this, all thermodynamic properties can be calculated by combinations of derivatives of $a_0$ and $a_r$ with respect to $\tau$ and $\delta$. For example, the pressure can be determined by equation 2.13. Similarly other thermodynamic properties of interest can be computed using the fundamental equations and their derivates. For details regarding the solution of $\alpha^0(\tau,\delta)$ and $\alpha^r(\tau,\delta)$, the reader is directed to [18] .

$$p(T,\rho) = -\left(\frac{\delta a}{\delta v}\right)_T \tag{2.13}$$

### 2.1.3. Thermo-physical Libraries

As discussed in section 2.1.1 and 2.1.2, the ideal gas EOS has a limited applicability. For non-ideal fluid flow conditions dominated by z≠1, the ideal gas EoS gives inaccurate results for thermo-physical properties. To correctly compute the theromo-physical properties in such regions, it is required to use EoS to correctly predict the fluid behaviour. Different EoS are used through different thermo-physical libraries such as *CoolProp* and *FluidProp*, which have a wide application and usage.

#### FluidProp

*FluidProp* is a thermo-physical package which is developed by a group of researchers from the Propulsion and Power Group at the Delft University of Technology [5]. It is vastly used to model a wide variety of fluids and mixtures using different thermodynamic libraries enlisted in table 2.1. In applications such as CFD simulations and dynamic energy system modelling, the derivatives of thermodynamic properties are used. A large set of these derivatives can also be obtained using *FluidProp* [1].

*FluidProp* can be used for the following calculations:

- Accurate calculation of fluid properties in the vicinity of the critical point
- Computation of the critical point for mixtures of fluids
- Obtaining efficient algorithms for phase equilibria.

A native model for the SW-EoS is implemented in the *FluidProp* database. The SW-EoS forms the basis for the generation of the thermodynamic tables which are used in the LUT method for thermodynamic property computation.

[1] http://www.asimptote.nl/software/fluidprop

| Library | Model/Institute |
|---|---|
| *GasMix* | Ideal Gas Equation |
| *RefProp* | National Institute of Standards and Technologies |
| *StanMix* | improved Peng Robinson cubic equation of state Stryjek Vera modification |
| *PCP-SAFT* | Perturbed Chain Polar SAFT equation of state |
| *IF97* | International Association for the Properties of Water and Steam |
| *TPSI* | Department of Mechanical Engineering, Stanford University |

Table 2.1: Thermodynamic libraries in FluidProp

#### CoolProp

CoolProp[2] is an open source thermo-physical fluid properties database that contains pure fluids, pseudo-pure fluids, and properties for humid air. The thermodynamic properties in *CoolProp* are calculated using the Helmholtz energy EoS, also known as the multi-parameter equations of state (MEOS) [19]. Almost all the fluid models in *CoolProp* are based on Helmholtz energy formulations, and therefore the thermodynamic properties of interest can be obtained directly from partial derivatives of the Helmholtz energy. It can provide an accurate calculation of thermodynamic properties such as enthalpy and entropy for over 100 fluids. However, a major limitation of *CoolProp* is its inability to compute thermo-physical properties for mixture of fluids [2]. *Coolprop* can be used across various platforms and has wrappers for the commonly used programming languages and technical software such as Javascript, Microsoft Excel, Python, Fortran, MATLAB etc [2].

## 2.2. Tools required for LUT implementation

### 2.2.1. Nearest neighbour search: k-d tree algorithm

The *Nearest Neighbour* (NN) search involves finding one or more data points which are closest to a given *query point*. The NN search is of practical importance in a number of fields, the most common of which involve data compression, image pixel analysis and data retrieval. In any LUT method, NN search is essential in identifying the correct index of the query vector or the grid element where it lies. The corresponding time associated with any search algorithm increases proportionally with the increase in number of mesh nodes or the size of the stored data tables. Using a fast binary search algorithm for grid location and nearest-neighbour search, a considerable reduction in search time and thereby, reduction in overall computational time can be achieved.

The k-d tree is a data structure for organizing points in a k-dimensional space. They are defined as a binary trees in which every node is a k-dimensional point [13] [3]. At every non-terminal node, the space is split into two halves by a splitting hyperplane. The left subtree consists of the points to the left of this hyperplane plane and the right subtree contains the points to the right of the hyperplane.The initial direction of split can be chosen at random. For example, if x is chosen as the initial dimension, then the initial splitting hyperplane will be along the y axis, in case of a 2-d tree [13] [3]. This hyperplane will pass through the median x value of the initial data set. All x values smaller than the x value on the splitting plane will be in the left sub-tree and all x values larger than the x value on the splitting plane will be in the right sub-tree. This procedure is recursively followed till every sub space contains only one data point [13] [3].

For the LUT method discussed in chapter 3, k-d tree algorithm is used to find the location of the given thermodynamic pair (query vector) on the thermodynamic domain (grid) and the nearest neighbours.

[2] www.coolprop.org

### 2.2.2. Data Interpolation

The set of thermodynamic quantities normally involved in the process or fluid-dynamic calculations, e.g., $P, T, h, c, c_v, c_P$, can be calculated using an interpolation method applied to thermodynamic tables [15] [7] [12] [20]. Interpolation methods like polynomial interpolation (linear, cubic, etc.) or Inverse data interpolation can be easily employed on the LUTs to obtain the desired thermodynamic quantities.

### 2.2.3. Shephard's Interpolation

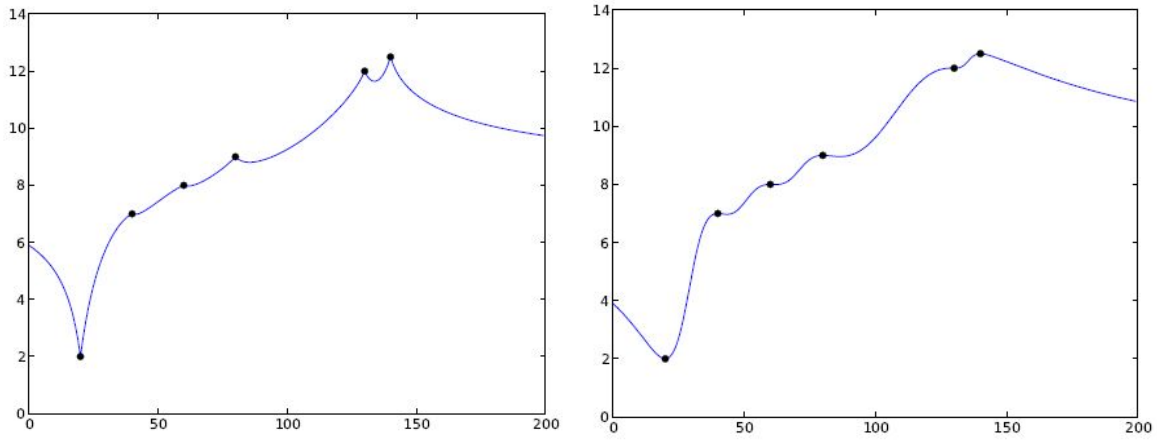
Shephard's method is a type of Inverse Distance Weighting (IDW) method. It is a deterministic method for multivariate interpolation with a set of known scattered data points. With the help of the weighted average of the values at known data points, the values at the unknown points are calculated [10] . It is a type of IDW since it makes use of the inverse of the distance of the query point to each known data point (proximity) while assigning weights. The interpolated function is given as:

$$\widetilde{f}(x) = \sum_{k}^{N} \frac{w_k(x)}{\sum_{n=1} w_i(x)} f(x_k) \tag{2.14}$$

In equation 2.14, $w_i$ is the weight function at a point $i$ and is given by:

$$w_i(x) = ||x - x_i||^{-p} \tag{2.15}$$

In equation 2.14 and 2.15 , $||x - x_i||$ denotes the distance between the query point and the data point and $p$ is a positive real number. As mentioned by Ken Anjo et. al., it can be seen that as weight function will decrease as the distance between query point and data point increases (since $p$ is positive). Therefore, higher the value of $p$, more is the influence of the data points closer to the query point [10].




(a) p = 1 (b) p = 2

Figure 2.2: Shephard's interpolation for different values of p

For varying values of $p$, it can be noted that:

1. For $0 < p \leq 1$, $\widetilde{f}$ has sharp peaks. This can be seen in figure 2.2a where the curve shows sharp peaks at the data points [10] .

2. For $p > 1$, the interpolated curve is smooth at the data points [10] . However, it must be noted that the derivate values at these points will be 0. This is evident from figure 2.2b .

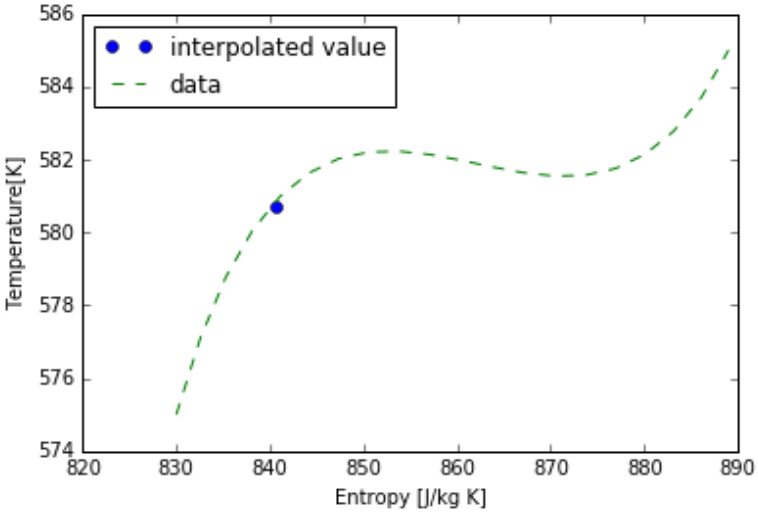



Figure 2.3: Exampale of Shephard's interpolation for randomized Temperature data

Although the weight functions can be controlled by the user using the parameter $p$, it can be seen from figure 2.2 that Shephard's interpolation might not be the best interpolation method if $p$ is not chosen carefully, keeping in mind the data points. An example of shephard's interpolation on randomized values in a temperature interval is shown in figure 2.3.

### 2.2.4. Polynomial Interpolation

*Polynomial interpolation* is the type of interpolation in which the aim is to construct a polynomial $p(x)$ using the values $f(x_0)$,$f(x_1)$,$(x_2)$....$f(x_n)$ of the function $f(x)$ at the known points $x_0$,$x_1$,$x_2$,...,$x_n$.If the interpolating polynomial is in the form of equation 2.16, then equation 2.17 means that $p$ interpolated the given data points.

$$p(x) = a_n x^n, a_{n-1} x^{n-1}, .., a_0 \qquad \text{with} \qquad n = [0,1,2...n] \tag{2.16}$$

$$p(x_i) = f_i, i = 0,1,2,...,n \tag{2.17}$$

Since there are $n+1$ conditions, we can have $n+1$ coefficients of $p$ which can be used to generate the required polynomial.
The simplest case of polynomial interpolation is when $n = 1$. This is the case of *linear interpolation*. The linear interpolant $p$ is expressed in the following ways:

$$p(x) = \frac{x - x_1}{x_0 - x_1}.f_0 + \frac{x - x_0}{x_1 - x_0}.f_1 \tag{2.18}$$

$$p(x) = f_0 + \frac{f_1 - f_0}{x_1 - x_0}.(x - x_0) \tag{2.19}$$

Equation 2.18 is the *lagrange's form* of linear interpolation whereas equation 2.19 is of the *Taylor's form*. Interpolating to just the function values is termed as *Lagrange interpolation*. Interpolating to function and consecutive derivative values of some function is termed as *Hermite interpolation*.
Figure 2.4 shows the interpolated function $p$ between points $x_0$ and $x_1$ of the function $f(x)$. The function values at $x_0$ and $x_1$ are $f_0$ and $f_1$ respectively. It can be noted here that at any given point $x_i$ it is possible that $p(x_i) \neq f_i$. This leads to a *interpolation error*. The interpolation error often depends upon the the type of data being interpolated and the interpolating function.

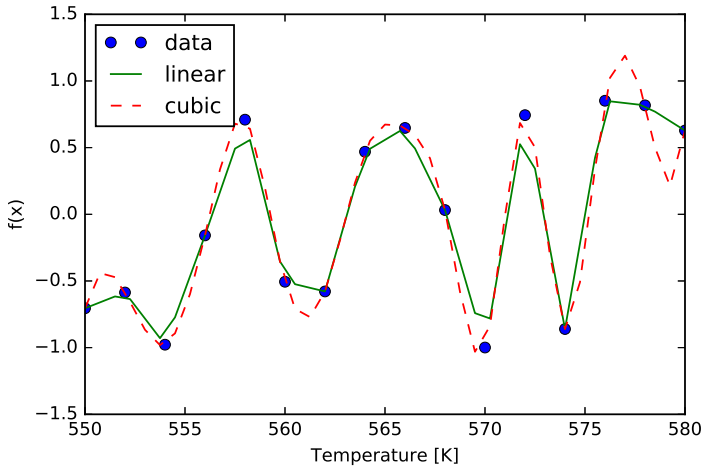



Figure 2.4: Exampale of linear interpolation and cubic interpolation for Temperature data points in 1 dimension

Equations 2.16 through 2.19 can be written for any polynomial of degree $n$ and can be modified to write in two or more variables. For instance for a cubic polynomial in two variables, equation 2.21 is used to interpolate the polynomial $p(x, y)$.

$$p(x,y) = \sum_{j=0}^{n} \sum_{i=0}^{m} a_{ij} x^i y^j \tag{2.20}$$

In a similar way polynomials of different degrees or number of variables can be generated.

Bicubic Interpolation
*Bi-cubic interpolation* is a form of polynomial interpolation for interpolating data points on a two-dimensional grid. The interpolated values are more accurate than those obtained by bilinear interpolation or nearest-neighbour interpolation on the same grid. For bi-cubic bivariate interpolation, the equation 2.20 can be written as follows:

$$p(x,y) = \sum_{j=0}^{3} \sum_{i=0}^{3} a_{ij} x^i y^j \tag{2.21}$$

For For bi-cubic, bi-variate polynomial interpolation, sixteen coefficients ($a_{i,j}$), are obtained using the function values $p(x, y)$ at sixteen known data points $(x, y)$.

### 2.2.5. Thermodynamic Consistency

For a given pure substance or a mixture of a certain composition, thermodynamic consistency implies that for three given properties (thermodynamic triple), for example $p, T$ and $\rho$, if $T = f(p,\rho)$ & $P = g(T,\rho)$, then $P \equiv g(f(P,\rho),\rho)$ [15]. In most of the existing LUT approaches, $f, g$ are replaced by their approximate counterparts i.e $\widetilde{f}, \widetilde{g}$ respectively. This replacement leads to the condition $P \neq g(f(P,\rho),\rho)$. This leads to a certain consistency error $\epsilon = P - \widetilde{g}(\widetilde{f}(P,\rho),\rho)$ [15]. In LUT applications, the thermodynamic consistency error can be reduced by a more accurate implementation of the method.

# 3

# Look Up Table Method

*This section describes in detail the application of the Look Up Table (LUT) method to calculate the thermodynamic properties of a fluid for given thermodynamic conditions.*

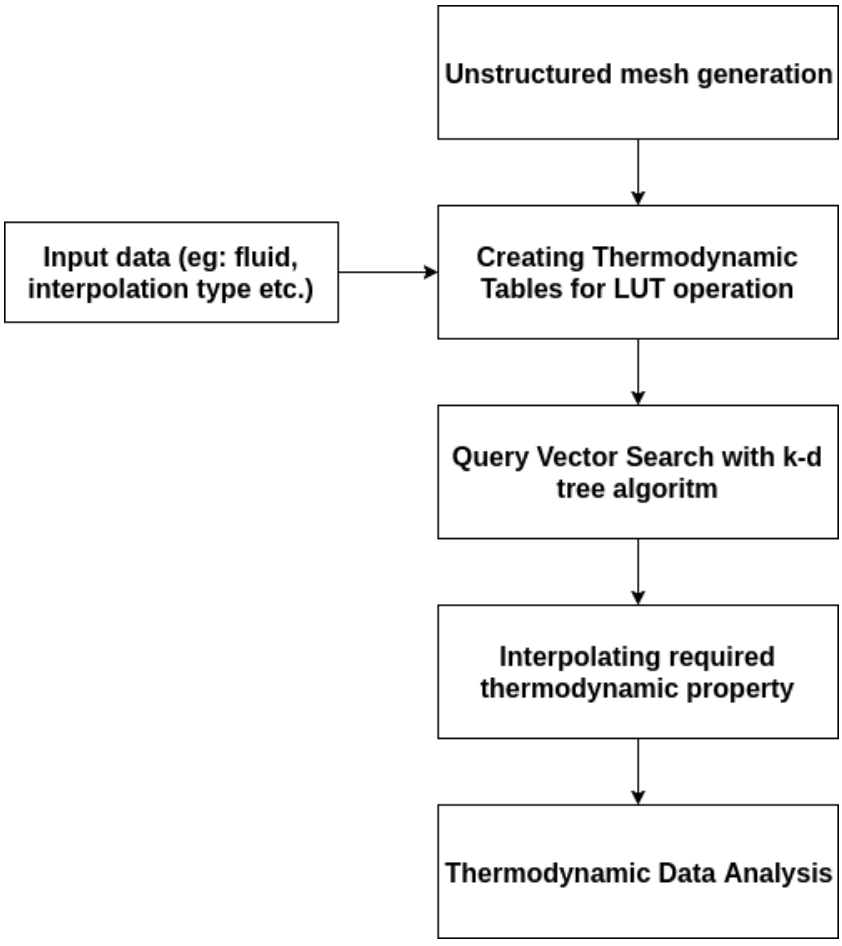



Figure 3.1: Flowchart showing the step by step procedure to calculate thermodynamic properties using the LUT Method

As discussed in chapter 2, to reduce the computational time, it is inherent that while obtaining thermo-physical properties for real fluid flows, an approximate property calculation technique should be used instead of direct solution of the complex EoS [16] [15]. The LUT methods are computationally less costly than solving the standard EoS when it comes to calculating the thermo-physical properties. The LUT algorithm combines

the array indexing operations with tools such as a binary search algorithm and an interpolation technique to calculate the thermo-physical properties. First, an unstructured mesh is generated on a thermodynamic domain of interest. The prerequisite data for construction of the initial thermodynamic tables is either obtained from pre-existing experimental data or through EoS libraries like *FluidProp* [5] or *Coolprop* [2]. This data is stored on each node of the grid. The query point(s) of interest for which the thermo-physical properties are desired is then located on the grid using a search algorithm. Lastly, the properties are computed using an interpolation technique. Figure 3.1 shows the basic structure of the LUT approach for thermodynamic property calculation.

## 3.1. Unstructured Thermodynamic mesh generation

In an unstructured grid, all nodes are automatically defined in an arbitrary manner. The elements of the mesh are triangles or tetrahedrons for 2D and 3D meshes respectively. Since the unstructured meshes are flexible and fast, they are suited to mesh complex geometries.

The first step towards the thermodynamic mesh generation for the proposed LUT method is the discretization of the saturation lines for the given fluid. The saturation lines for the fluid are discretized according to a given time interval. The discretized liquid and vapour saturation lines for octamethyltrisiloxane ($C_8H_{24}O_2Si_3$) (MDM) is presented in Figure 3.2. The discretized points for the construction of the geometry for the VLS curve can be placed both uniformly as well as variably. It can be seen in figure 3.2 that the points are placed uniformly on both vapour and liquid saturation lines, till they approach the critical point. Near the critical point, the density of points on the vapour and liquid saturation curves is increased to capture the geometry of the curves in the critical region more accurately. The curve is represented in normalized domain with the help of an algebraic transformation. Figure 3.3a shows how the single phase region is incorporated as a separate zone. Figure 3.3b shows the unstructured thermodynamic mesh generated by UMG2. Each node of this mesh is used to store the thermo-physical properties obtained from the thermodynamic library *FluidProp*.

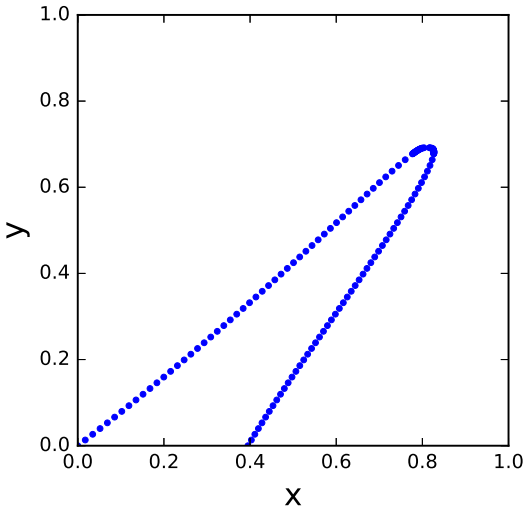
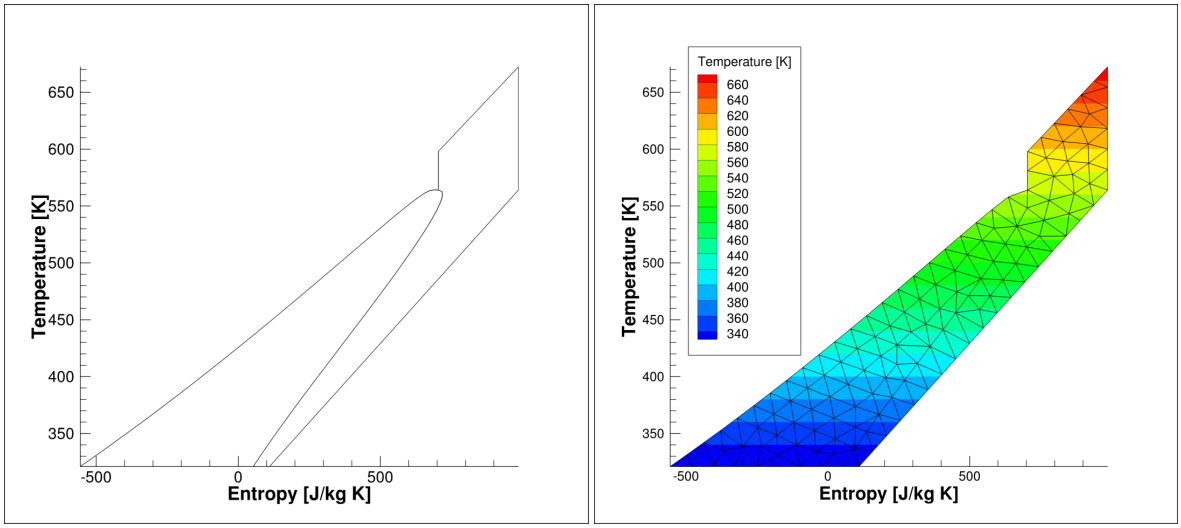



Figure 3.2: Discretized liquid and vapour saturation curves

### Multi-zone Mesh Generation

In figure 3.3a, it can be seen that the geometry extends to also include the supercritical zone. This is important to represent the processes that occur in the dense gas region. Across the saturation curves, the properties such as speed of sound exhibit high discontinuities. Therefore to avoid high interpolation errors, the grid is meshed as 2 separate zones, one for the VLE region and the other for the single phase region. The discretized saturation curves form the base for meshing the 2 zones separately. Figure 3.3a shows that the domain is divided into 2 zones over which the mesh is generated. These zones have their own separate mesh connectivities. This is also greatly advantageous in the location of the query vector and its neighbours as the spatial




(a) Division of zones on the mesh

(b) Unstructured mesh for MDM

Figure 3.3: Unstructured Thermodynamic mesh generation for the LUT Method

search can be exclusively conducted in either of the zones separately.

### UMG2

The mesh generation tool *UMG2* is an automatic tool for mesh generation on geometries of an arbitrary, complex shapes for unstructured forms. The geometry of the domain and mesh element sizing information is provided to UMG2 in the form of data input files. The triangular elements of the mesh are generated based upon delauney's triangulation method. This process is entirely automated by UMG2. The functionality of *UMG2* is beyond the scope of this work.

## 3.2. Thermodynamic Table Generation

The runtime computation for various processes is replaced with simple array indexing operations on stored data tables using an LUT approach. These tables can be pre-calculated and stored in the program memory, calculated as part of a program's pre-processing phase, or they can even be stored in hardware for application-specific platforms.

For the LUT method presented here, for a given fluid, the thermodynamic tables are generated for each block (zone) of the mesh as a part of pre-processing. The thermodynamic table stores the thermodynamic properties corresponding to every node of the mesh. The thermodynamic properties for each node are obtained using the thermodynamic library *FluidProp* [5] which has a native model for SW-EoS implemented in its database. The thermodynamic tables are stored in the program's memory after the initial generation.

The thermodynamic tables form the basis of operation for the k-d tree search algorithm and the thermodynamic interpolation. As a result of the k-d tree operation on the thermodynamic tables, the nearest neighbours for the the desired *query vector* are obtained. The thermodynamic properties of the nearest-neighbour points are used to interpolate the thermodynamic properties at the query vector.

## 3.3. Query vector and nearest neighbour search

As noted by Pini et. al. [15], for a given *query vector* in any thermodynamic input pair, the identification and location of the correct grid element which contains the query vector, underlines one of the most important operations of any LUT method [15]. The corresponding time associated with a search algorithm increases proportionally with the increase in number of mesh nodes. Using a fast binary search algorithm for grid location and nearest-neighbour search, a considerable reduction in search time and thereby, reduction in overall computational time can be achieved. For the LUT method presented, a robust kd-tree space partitioning algorithm is implemented. The thermodynamic domain is transformed into a tree structure with the help of the k-d tree algorithm for accurate and efficient search procedure. The size of the tree structure depends upon the number of nodes in the grid. Since the k-d tree binary search algorithm functions on the Euclidean

space for faster search, the entire thermodynamic domain is transformed into a normalised space. The k-d tree search algorithm is used in the LUT algorithm for the following two purposes:

1. **Locating the query vector on the grid:** The k-d tree algorithm retrieves the element on the grid node on which the query vector lies.

2. **Locating the nearest-neighbours to the query vector:** The k-d tree search also retrieves the desired number of points nearest to the query vector. The search algorithm obtains the values of the index (node) of the required number of points to the query vector. This search is based upon the euclidean distance of the search points from the query vector. Once the index of the nodes nearest to the query vector is known, the thermodynamic properties at those nodes can be obtained using the thermodynamic tables which are generated as part of section 3.2. These properties are then used as a part of section 3.4 to obtain the interpolated values at the query vector. An example of the euclidean distances and the grid element in which query vector location and its nearest neighbour search on the thermodynamic domain is presented in table 3.1. The algorithm identifies that the query vector lies on element number 2909.

| | Zone | Element | Euclidean distance (from query vector) |
|---|---|---|---|
| Query vector | 2 | 2909 | 0.00 |
| NN 1 | 2 | 2913 | 0.009 |
| NN 2 | 2 | 2910 | 0.0098 |
| NN 2 | 2 | 2911 | 0.0099 |

Table 3.1: Location of the query vector and 3 nearest neighbours as identified by the k-d tree search algorithm

## 3.4. Thermodynamic Interpolation

As a part of the final step for the thermo-physical property computation, the application of an interpolation algorithm is required. The LUT method presented here uses two interpolation methods, namely, *bicubic bivariate interpolation* and *inverse data weighing*, in the form of *shephard's interpolation.*

#### Shephard's Interpolation:

The Shephard's interpolation scheme, as described in section 2.2.2 is a deterministic method suitable for multivariate interpolation. For the most optimum results, the number of scattered data points for interpolating the required thermodynamic property is set as $n = 3$. Shephard's interpolation can be implemented for a different set of input states like $(p,T)$, $(p,s)$, $(v,s)$, $(h,s)$ and $(v,u)$.

#### Bi-cubic Bivariate Interpolation:

Bi-cubic bivariate interpolation uses a bi-cubic function to interpolate the thermodynamic properties of interest using the thermodynamic tables. According to equation 2.21 , 16 values of coefficients $a_{ij}$ are required for interpolation. These coefficients are calculated using the thermodynamic data obtained from the 16 nearest neighbours of the query vector with the help of the $k-d$ tree algorithm, as described in section 3.3. The $k-d$ returns the index and the euclidean distance of the nearest 16 data points from the query vector. The thermodynamic property of interest is then computed with the help of a cubic function explained in section 2.2.2 .

## 3.5. Process Automation

For the different steps involved in the process of thermo-physical property calculation through the LUT tool, interfacing between multiple tools (fortran codes) is required. Figure 3.4 demonstrates the automation process for interfacing between different tools to determine the thermo-physical properties and then post-process them for analysis. The tool *tmesh* obtains the thermodynamic data from *FluidProp* and using that data determines the geometry of the VLS curve for the selected working fluid on the thermodynamic plane of interest. This geometry of the VLS curve is then passed onto *UMG2* for mesh generation. The mesh file is then used by the LUT tool. Along the mesh file, an input file is also passed onto LUT tool. This input files

store data such as the thermodynamic input pair (eg: $(v,s)$), interpolation type, search algorithm type, etc. The thermo-physical property calculation takes place within the main LUT tool which has the binary search and interpolation algorithms implemented within itself. The thermo-physical properties are mesh data are then written in data files which can be analysed for mesh sensitivity, accuracy and computational time.

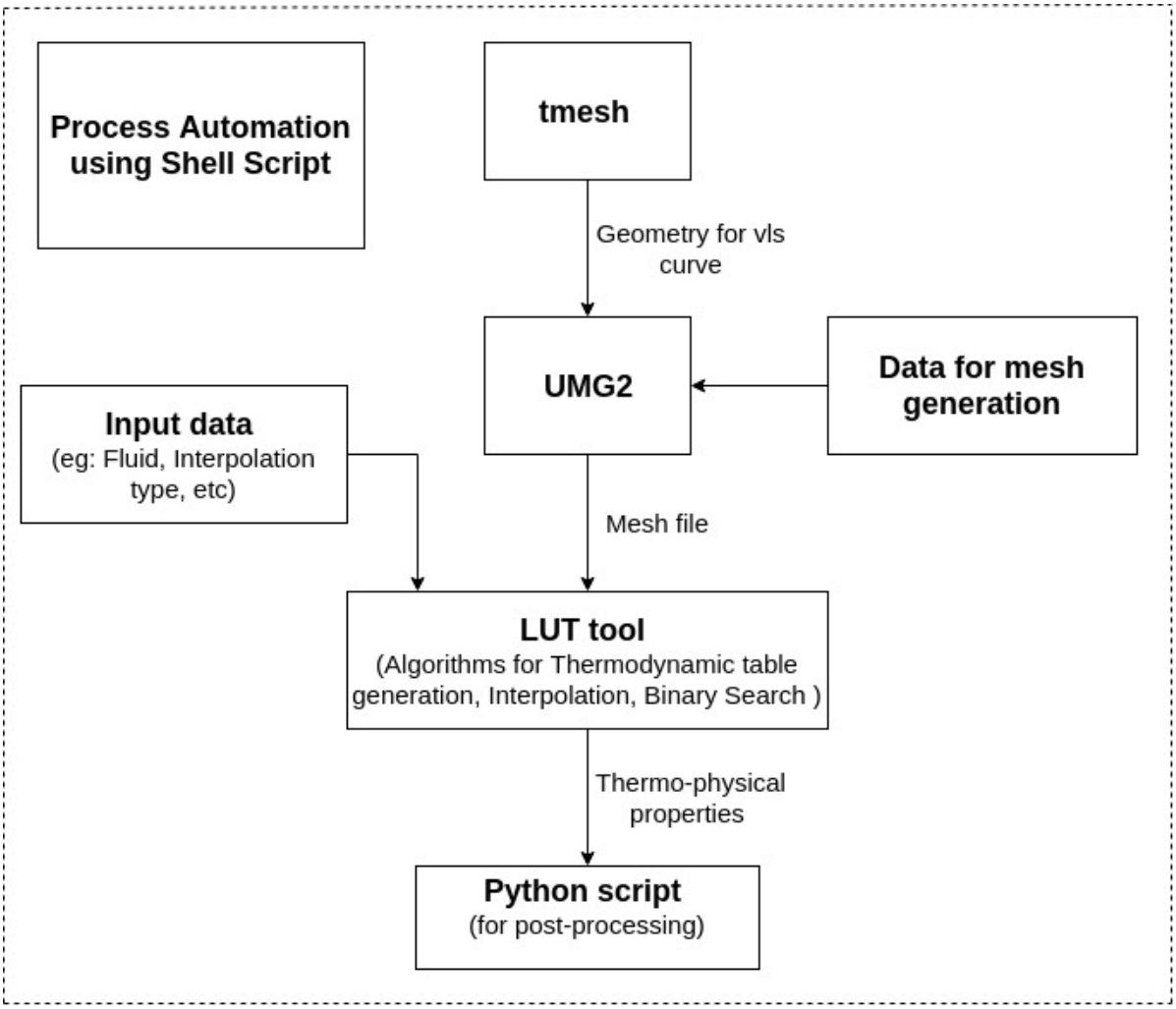



Figure 3.4: Flowchart showing the process automation for generation and analysis of thermo-physical properties

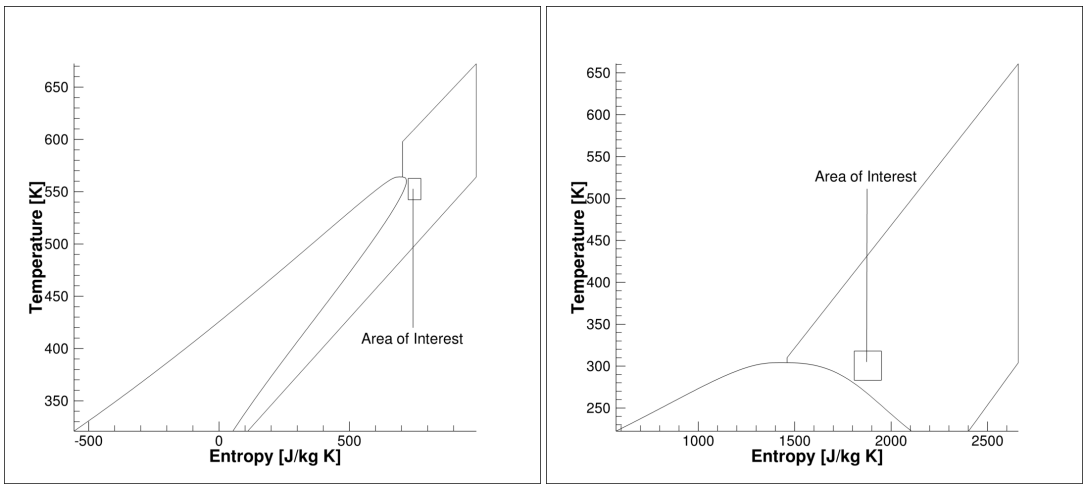
# 4

# Analysis and Results of the LUT Method

*In this chapter the computational efficiency and accuracy of the LUT method for thermo-physical property calculation are assessed.*

## 4.1. Computational Time

The major advantage of using the LUT method is the gain in computational efficiency while calculating thermo-physical properties of the given fluid. In this chapter, the computational efficiency and accuracy of the LUT method are analysed for Siloxane MDM and Carbon Dioxide (CO2). The thermodynamic tables are generated based upon the Span-Wagner EoS implemented in *FluidProp*.




(a) $(T-s)$ thermodynamic domain for MDM (b) $(T-s)$ thermodynamic domain for CO2

Figure 4.1: Thermodynamic domains for MDM and CO2 considered for analysis

Figure 4.1 shows the thermodynamic domains considered for MDM and CO2. The compressibility factor $z$ varies between 0.7 & 0.9 for MDM and for CO2, the value of $z$ varies between 0.7 & 0.8. For the analysis, the areas of interest, as shown in figure 4.1 are chosen to analyse the computational cost and accuracy for thermo-physical property calculation close to the critical point where non-ideal fluid behaviour is observed. The areas of interest for both fluids is isolated and meshed separately. For the analysis, 10000 randomly distributed query points are considered on the $(T-s)$ thermodynamic domain, near the critical region. For various thermodynamic input pairs, namely $(h,s)$, $(vs)$,$(pT)$,$(ps)$, a set of thermodynamic properties, namely $(p,t,h,s,c,u,c_v,c_p)$, is evaluated using the LUT method on in the area of interest. The number of nodes on the grid is varied between 150 and 4785 for the analysis.

The computational efficiency of the LUT method is now analysed by comparing the computational time for the LUT method with the time for direct evaluation of thermodynamic properties using *FluidProp*. Tables 4.1 and 4.2 show the ratios of computational times of the two interpolation methods with respect to *FluidProp* for MDM and CO2 respectively. It can be seen that both interpolation methods used in perform better than the direct property evaluation using the EoS in *FluidProp*. For Shephardś interpolation, maximum gain in computational efficiency can be seen for the thermodynamic input pair $(p,s)$, which is 33.8 times faster than direct property calculation using EoS. For bi-cubic interpolation, the gain in computational efficiency for the same input pair with respect to direct property calculation is 22.61 times over the direct property calculation using EoS. A similar reduction in computational cost can be noticed for other input pairs. Shephardś interpolation method proves to be faster than bi-cubic interpolation because it requires the search algorithm to be performed for 3 nearest neighbours as opposed to 16 for bi-cubic interpolation. This gain is also attributed towards the simpler interpolation algorithm which does not require the calculation of interpolating coefficients, as required by bi-cubic interpolation.

| | $vs$ | $pT$ | $hs$ | $ps$ |
|---|---|---|---|---|
| Time(*FluidProp*)/Time(Shephard) | 32.21 | 33.37 | 32.98 | 33.81 |
| Time(*FluidProp*)/Time(Bicubic) | 21.95 | 22.35 | 22.22 | 22.61 |

Table 4.1: Comparison of the ratio of times between *FluidProp* and LUT method (Shephard's interpolation and Bi-cubic interpolation) fr different input pairs for MDM for 4785 nodes

| | $vs$ | $pT$ | $hs$ | $ps$ |
|---|---|---|---|---|
| Time(*FluidProp*)/Time(Shephard) | 38.97 | 39.22 | 39.13 | 39.00 |
| Time(*FluidProp*)/Time(Bicubic) | 23.64 | 25.82 | 24.89 | 24.43 |

Table 4.2: Comparison of the ratio of times between FluidProp and LUT method (Shephard's interpolation and Bi-cubic interpolation) for different input pairs for CO2 for 4785 nodes

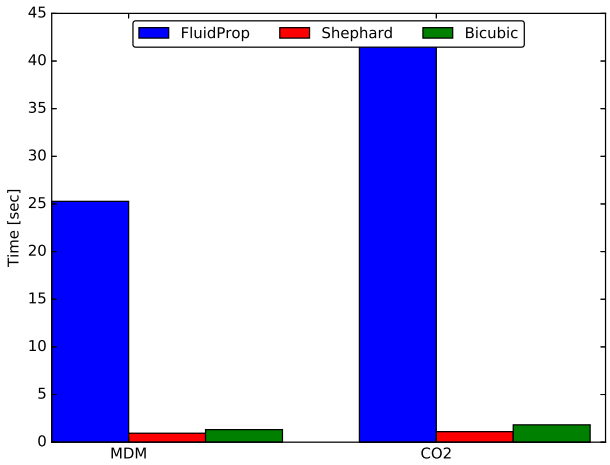



Figure 4.2: Comparison of computational time for *FluidProp*, Shephard's interpolation and Bi-cubic non consistent interpolation for MDM and CO2 for 4785 mesh nodes

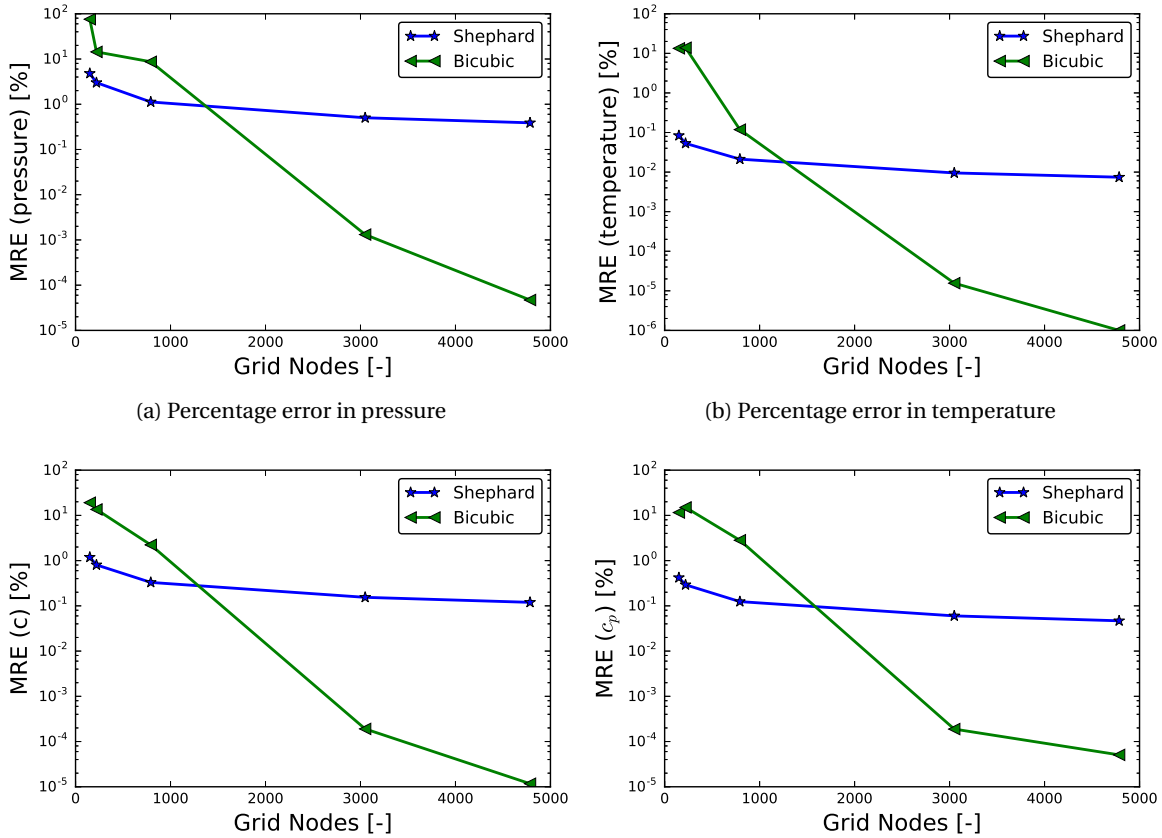
A large gain in computational efficiency for both MDM and CO2 can be seen through figure 4.2. The shephardś interpolation method is 26.8 times faster than *FluidProp* for MDM and 38.9 times faster than *FluidProp* for CO2. The bi-cubic interpolation is 24.2 times faster than *FluidProp* for MDM and 23.8 times faster than *FluidProp* when the working fluid is CO2. The gain in computational efficiency for LUT method (for

both shephardś interpolation and bi-cubic interpolation) can be attributed to the replacement of direct property calculation by array indexing operations (binary-searching and interpolation). Shephard's interpolation proves to be the fastest for both MDM and CO2 because of its simple inverse data weighing method used for interpolation. Bi-cubic interpolation proves to be slower than shephardś method because of the calculation of coefficients required for bi-cubic interpolation. Overall, the LUT method outperforms the direct property calculation by *FluidProp*.

## 4.2. Accuracy

The accuracy of LUT method is hereby evaluated quantitatively by computing the mean relative error in the thermodynamic properties against the quantities computed by the Span Wagner EoS implemented in *FluidProp*. Similar to section 4.1, 10000 points are considered for assessment in the region of interest shown in figure 4.1a and 4.1b for MDM and CO2 respectively. The results in this section report the Mean Relative Error (MRE) for the set of thermodynamic properties ($p, t, c, c_p$...) for the input pair ($v, s$). For any given thermodynamic property $\phi$, the $MRE$ is evaluated by equation 4.1, as below:

$$MRE\% = \sum_{i}^{n} \frac{\phi_{LUT} - \phi_{FP}}{\phi_{LUT}} 100 \tag{4.1}$$




(a) Percentage error in pressure

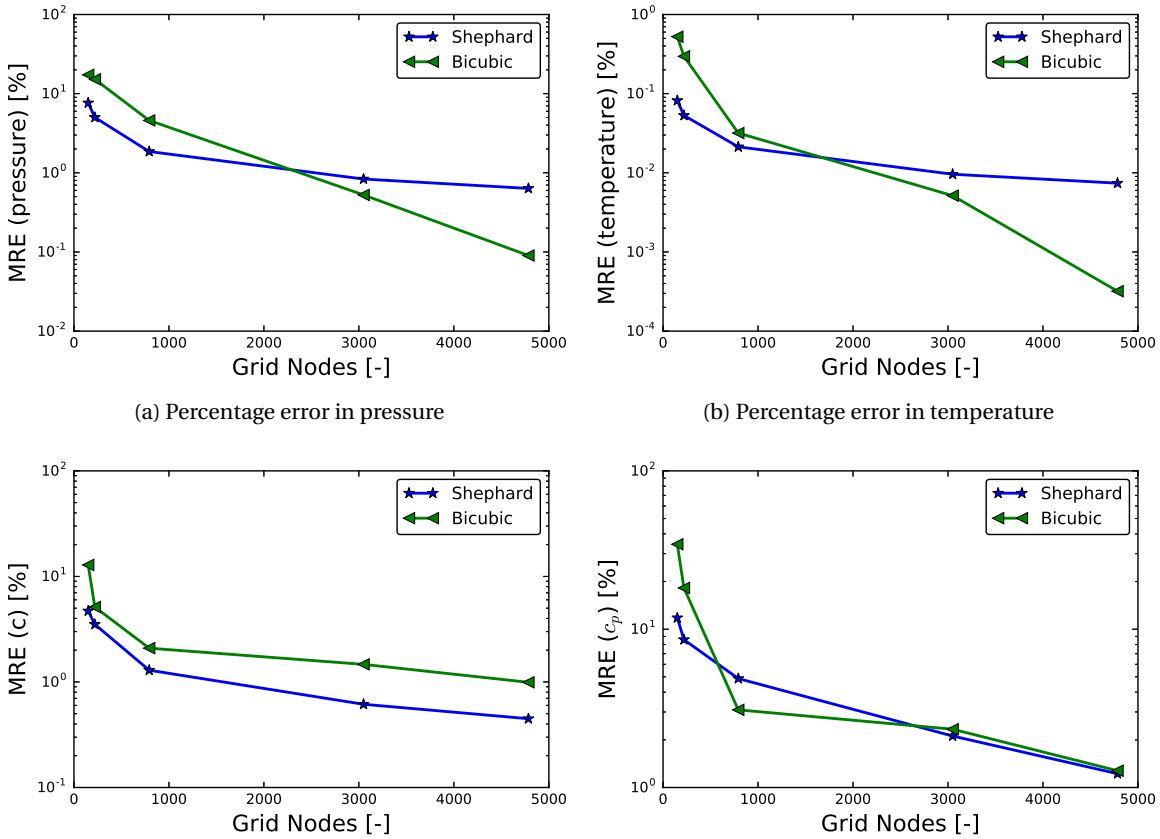
(b) Percentage error in temperature

(c) Percentage error in speed of sound ($c$)

(d) Percentage error in specific heat ($c_p$)

Figure 4.3: Variation of Mean Relative Error [%] in thermodynamic properties of 10000 randomly distributed query points for MDM

Figure 4.3 shows that for coarse grids, both Shephard's interpolation and bi-cubic interpolation have relatively high MRE. This is because of a large variation in data points used to interpolate the thermodynamic values. However as the grid becomes finer, both interpolation methods show an increase in the accuracy of the thermo-physical properties. For MDM, the errors in pressure reduce from 4.8% for 150 nodes to 0.3% as the node size is increased to 4785. A similar trend can be noticed for other thermodynamic properties. The bi-cubic interpolation outperforms the Shephard's interpolation for finer meshes with errors of upto 4 orders lesser than shephardś interpolation.




(a) Percentage error in pressure

(b) Percentage error in temperature

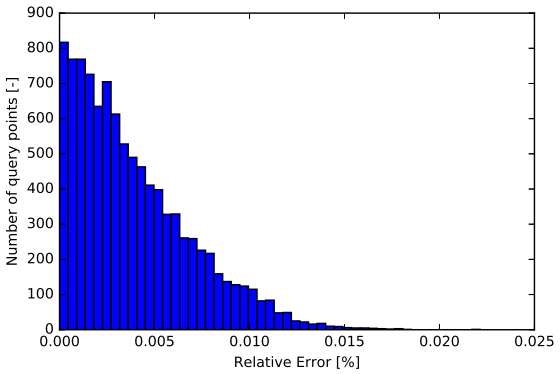
(c) Percentage error in speed of sound ($c$)

(d) Percentage error in specific heat ($c_p$)

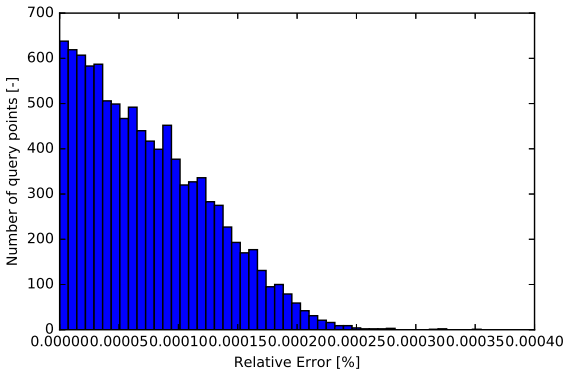
Figure 4.4: Variation of Mean Relative Error [%] in thermodynamic properties of 10000 randomly distributed query points for CO2

For CO2, although the overall observations are the same as those for MDM, however, it can be noticed from figure 4.4 that for finer meshes, Shephard's interpolation outperforms bi-cubic interpolation for calculation of $c$ and $c_p$. Overall, both interpolation methods show a significant improvement in accuracies as meshes become finer.

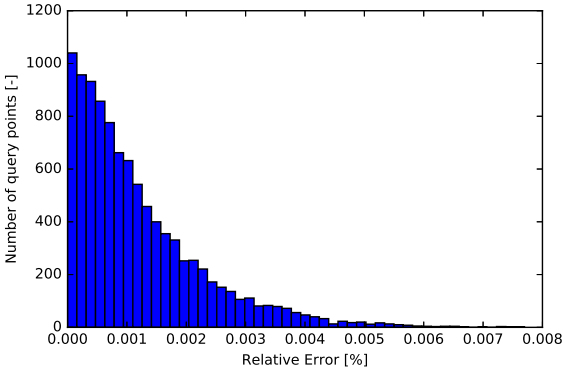
Figure 4.5 shows the frequency distribution of 10000 query points for absolute error in thermodynamic properties. For the frequency distribution in pressure, the frequency of query points is higher for smaller relative error values. Under 0.05% of the query points show the relative error over 0.015% in pressure and over 0.00025% in temperature. 99.98% points have an error in speed of sound less than 0.007% and the same percentage of points show less than 0.0005% error in $c_p$. Similar observations can be made for CO2 from figure 4.6. Frequency distribution for $c$ and $c_p$ for CO2 have been adjusted to exclude the 0.3% points with error more than 0.005% and 0.01% points with error more than 0.0040% for $c$ and $c_p$ respectively.



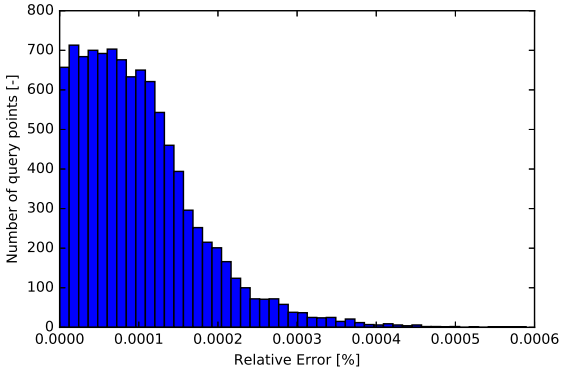

(a) Frequency distribution for absolute error [%] in pressure




(b) Frequency distribution for absolute error [%] in temperature

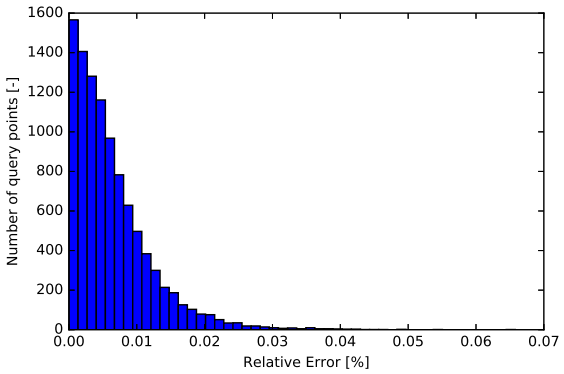



(c) Frequency distribution for absolute error [%] in speed of sound ($c$)

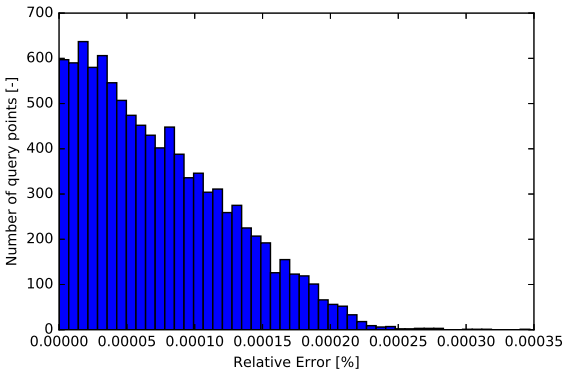



(d) Frequency distribution for absolute error [%] in specific heat capacity ($c_p$)

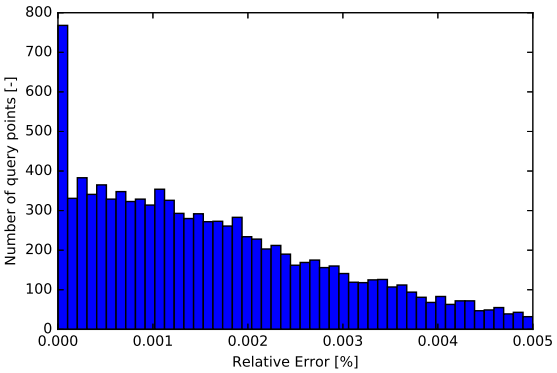
Figure 4.5: Frequency distribution of points for the absolute error [%] in thermodynamic properties calculated using Shephard's interpolation with 4785 grid points for MDM



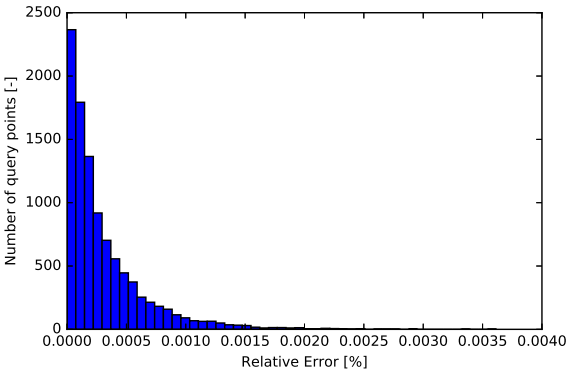

(a) Frequency distribution for absolute error [%] in pressure




(b) Frequency distribution for absolute error [%] in temperature




(c) Frequency distribution for absolute error [%] in speed of sound ($c$)




(d) Frequency distribution for absolute error [%] in specific heat capacity ($c_p$)

Figure 4.6: Frequency distribution of points for the absolute error [%] in thermodynamic properties calculated using Shephard's interpolation with 4785 grid points for CO2

# 5

# Application of LUT

*This chapter presents two applications of the Look Up Table method. The time comparison for property computation and the accuracy of the thermo-physical properties calculated using the LUT approach is presented for both applications.*

## 5.1. Isentropic Expansion through a fixed Control Volume

It is well known that for the isentropic expansion or compression of an ideal gas, the Mach number is only dependent upon the pressure ratio at the exit (static) to inlet (total) and on the specific heat ratio [6] . However, in some cases an expansion process can occur in a region where the ideal gas law is not applicable ($z \neq 1$). This section represents an application of the LUT method to compute the Mach number and pressure ratios of an isentropic expansion process occurring in the region where ($z \neq 1$).

### 5.1.1. Control Volume

For an inertial reference frame, a control volume is a volume fixed in space or moving with constant flow velocity through which the continuum (gas, liquid or solid) flows. Under steady state conditions, a control volume can be thought of as an arbitrary volume in which the mass of the fluid, gas or solid flowing through it, remains constant. As a continuum moves through the control volume, the mass entering the control volume is equal to the mass leaving the control volume. When no work is done on or by the system and there is no transfer of heat, the energy within the control volume remains constant. Often in propulsion and power problems, the dynamics within the control volume are of great interest. For any given fluid flow conditions, the basic laws of nature, that is, conservation of mass and energy are always satisfied. In addition to the flow conditions, each flow is subjected to certain physical constraints, mathematically referred to as boundary conditions, which must be satisfied physically [21].

#### Conservation of Mass

The mass rate change inside any control volume is given by the difference between the mass flow rate at the entry and the mass flow rate at the exit of the control volume. For a single inlet and a single outlet control volume, the mass flow rate is given as follows by equation 5.1 :

$$\frac{dm_{cv}}{dt} = \dot{m}_{in} - \dot{m}_{out} \tag{5.1}$$

For steady flow, there is no mass added to or subtracted from the system. Therefore, the mass flow rate at the inlet and the exit of the control volume is the same and is given by 5.2.

$$\dot{m}_{cv} = \dot{m}_{in} = \dot{m}_{out} \tag{5.2}$$

#### Conservation of Energy

The physical idea behind the energy flow through a control volume is that any rate of change of energy in the control volume can only be caused by the rates of energy flow into or out of the system. Since the heat transfer and the work done are already included, the only other contribution towards a change in energy of

the system can only be associated with the mass flow in and out of the system, which carries energy with it. This gives the desired form of the energy equation as follows:

$$\begin{aligned}\text{(Rate of change of energy in CV)} = \text{(Rate change of heat added to CV)} - \text{(Rate of work done)} \\ + \text{(Rate of energy flow into the CV)} - \text{(Rate of energy flow out of CV)}\end{aligned} \tag{5.3}$$

The first law for a fluid flow can thus be written as:

$$\frac{d}{dt}\sum E_{cv} = \sum \dot{Q_{cv}} + \sum \dot{W} + \sum \left(\dot{m}(h + \frac{v^2}{2}) + gz\right) \tag{5.4}$$

In equation 5.4, $\dot{Q}$ and $\dot{W}$ are often referred to as *thermal power* and *mechanical power*. For steady state assumption, these quantities can be assumed to be zero. Moreover, for aerospace applications the velocities are very high and the term that is associated with changes in the elevation or height is almost negligible in comparison. Therefore, the term $gz$ can be neglected. The simplified energy equation is given by 5.6 . For a detailed explanation on each term mentioned in the equations 5.3 and 5.4 , the reader is referred to [21].

$$0 = \left(h_{exit} + \frac{v_{exit}^2}{2}\right) - \left(h_{in} + \frac{v_{in}^2}{2}\right) \tag{5.5}$$

$$h_{tot} = \left(h_{exit} + \frac{v_{exit}^2}{2}\right) = \left(h_{in} + \frac{v_{in}^2}{2}\right) \tag{5.6}$$

### Isentropic Expansion through Control Volume

If the fluid expansion takes place with no flow of heat energy either into or out of the system, the process is called isentropic expansion. The relations of the pressure ratio and temperature ratio with the Mach number is given by equations 5.7 and 5.8 respectively. For the derivation of these relations, the reader is directed to [9].

$$\frac{P_t}{P} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{-\gamma}{\gamma - 1}} \tag{5.7}$$

$$\frac{T_t}{T} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{-1} \tag{5.8}$$

However, in the case of an expansion process taking place under non-ideal conditions, the ideal-gas law is not valid. For such thermodynamic states, where ideal gas law cannot be used to derive the isentropic flow relations, the isentropic gas flow can be computed using equations 5.6 and 5.2 . For an isentropic flow through a nozzle with a varying cross-section area, $A = A(x)$, these equations can be written as follows:

$$\dot{m}_x = (\rho A u)_x = constant \tag{5.9}$$

$$h_{0x} = (h + \frac{1}{2}u^2)_x \tag{5.10}$$

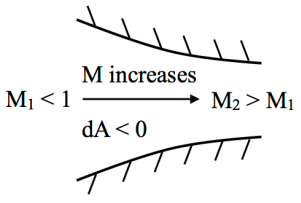



Figure 5.1: Single inlet and outlet Control Volume for subsonic flow conditions

Figure 5.1 shows a control volume through with a single inlet and outlet. The control volume can be treated as adiabatic (negligible heat transfer between the gas and CV wall), The system is assumed to be operating at steady state so that there are no occurring in mass of fluid or its energy within the control volume. The gravitational potential energy of the fluid is neglected. For an isentropic flow across a control volume

with single inlet and outlet, the total enthalpy remains constant across the process. This total enthalpy at any location in the system can be expressed as a function of the total pressure and temperature at the inlet of the control volume. The density and the static enthalpy can be expressed as functions of static pressure and entropy. Since the entropy of the system remains constant, it can also be expressed as a function of the total conditions at the inlet.

$$h_{0x} = h_{0x}(p_{t0}, T_0) \tag{5.11}$$
$$h_x = h_x(p, s) \tag{5.12}$$
$$\rho_x = \rho_x(p, s) \tag{5.13}$$
$$s = s(p_{t0}, T_0) = constant \tag{5.14}$$

Using equations 5.9 through 5.14, the energy equation an be reduced in the form of the following implicit function:

$$h_{0x}(P_{t0}, T_0) = \left( h(p, s) + \frac{1}{2} \left( \frac{\dot{m}}{\rho(p, s)} \right)^2 \right)_x \tag{5.15}$$

$$h_{0x}(P_{t0}, T_0) = f(p) \tag{5.16}$$

For given total inlet temperature, total inlet pressure, a fixed mass flow rate and area of the control volume at any given section of the geometry, equation 5.15 can be solved using an iterative root solving method like the Bisection Method.

To test the applicability of the LUT method, an isentropic expansion case is simulated through a control volume with MDM as working fluid on the $T - s$ thermodynamic domain with 4785 mesh nodes. The inlet conditions of the fluid are at the stagnation state (total condition). The control volume is discretized into 1000 equally spaced parts in the $x$ direction. The inlet and exit areas are fixed and the area varies linearly with the distance in the direction of the flow. The inlet conditions are shown in table 5.1 .

| **Parameter** | **Symbol** | **Value** | **Units** |
|---|---|---|---|
| Inlet to Exit area ratio | $A_r$ | 4 | - |
| Total inlet pressure | $p_{t0}$ | 1256847.5 | Pa |
| Total inlet temperature | $T_{t0}$ | 558.6 | K |
| Mass flow rate | $\dot{m}$ | 20.0 | kg/s |
| Compressibility factor | $z$ | 0.54 | - |

Table 5.1: Inlet conditions for the control volume

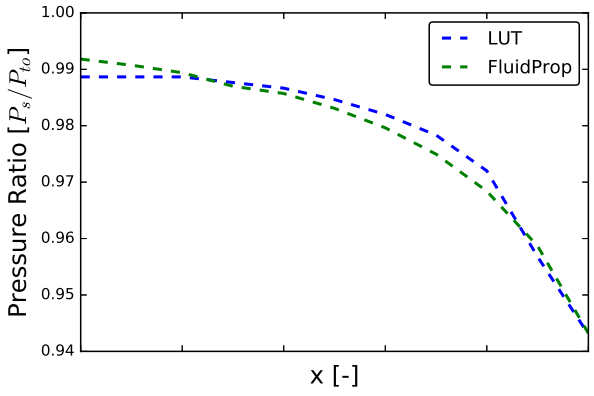



Figure 5.2: Variation in pressure ratio ($\frac{p_s}{p_{t0}}$) across the nozzle

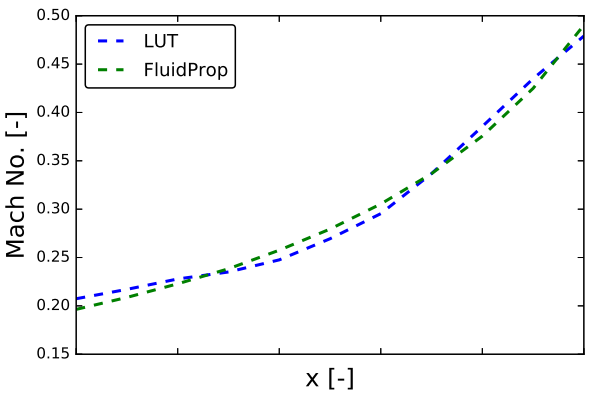



Figure 5.3: Variation of mach number through the nozzle

To obtain the static pressure ($p_s$) at any given position $x$, equation 5.16 can be implicitly solved. For the isentropic expansion process, the variation of static pressure is obtained for the discretied domain from inlet to exit of the nozzle. The solution of the equation 5.16 is obtained using the Bisection method for iterative root solving. To analyse the LUT method and test its accuracy, equation 5.16 is solved by using pressure values from *FluidProp* and Shephard's interpolation. An isolated grid with 4785 elements is used for the analysis.

| **MRE($p_s/p_{t0}$)** | **MRE(Mach)** | **Time(FluidProp/Time(LUT))** |
|---|---|---|
| 0.32 % | 0.21 % | 5.69 |

Table 5.2: Mean relative error in pressure ratio and mach number (with respect to FluidProp) and the time ratio between FluidProp and LUT

As seen in figure 5.3, an increase in number is noted and figure 5.2 shows the decreasing pressure ratio across control volumes while moving downstream in $x$ direction. As expected, LUT method with Shephard's interpolation is computationally less costlier than *FluidProp*. Table 5.2 shows the that the LUT method is approximately 5.69 times faster than the direct solution of EoS through FluidProp. This can be attributed to the reduction in time in LUT calculations, which is obtained because of direct search operation and numerical interpolation on existing thermodynamic tables instead of comparatively slower and complex EoS solution.

## 5.2. Supersonic Nozzle design through Method Of Characteristics

### 5.2.1. Method of Characteristics

The *Method of Characteristics* (MOC) is a technique for solving partial differential equations. The MOC technique can be used to design the for the supersonic nozzles. In this technique, partial differential equations are used to calculate isentropic properties for the supersonic flows. These supersonic flows belong to a hyperbolic class of partial differential equations. A set of characteristic equations and compatibility relations are used to obtain the supersonic nozzle profile (out of scope).

To test the applicability of the LUT method of thermodynamic property calculation, the LUT tool is integrated with an in house MOC tool, designed by Ir. Nitish Anand (Propulsion and Power group in TU Delft). For a detailed insight into the MOC technique for supersonic nozzle design through the MOC tool, the reader is referred to [1].

### 5.2.2. Data flow structure between LUT tool and MOC tool

To determine the nozzle design, the MOC tool requires certain thermodynamic properties for the working fluid under the desired working conditions. These properties can be obtained by the MOC tool using either the *Ideal gas equation* or *Van der Walls EoS* or *Refprop*. Integration of the MOC tool with the LUT algorithm provides another option for the MOC tool to obtain the thermodynamic properties of interest. To test the applicability of LUT tool for the supersonic nozzle design, the LUT tool is integrated with the MOC tool. For an input thermodynamic pair $(h, s)$ provided by the MOC tool, the LUT tool returns the thermodynamic data set $\{P, T, D, c, cp, cv\}$ for every iteration. Both MOC tool and LUT tool have their individual input files. The MOC and LUT input files are presented in appendix A. A file containing the mesh connectivity data for the thermodynamic domain is required for the thermodynamic tables to be created as a part of the pre processing. The overall data flow between the tools is represented in figure 5.4 .

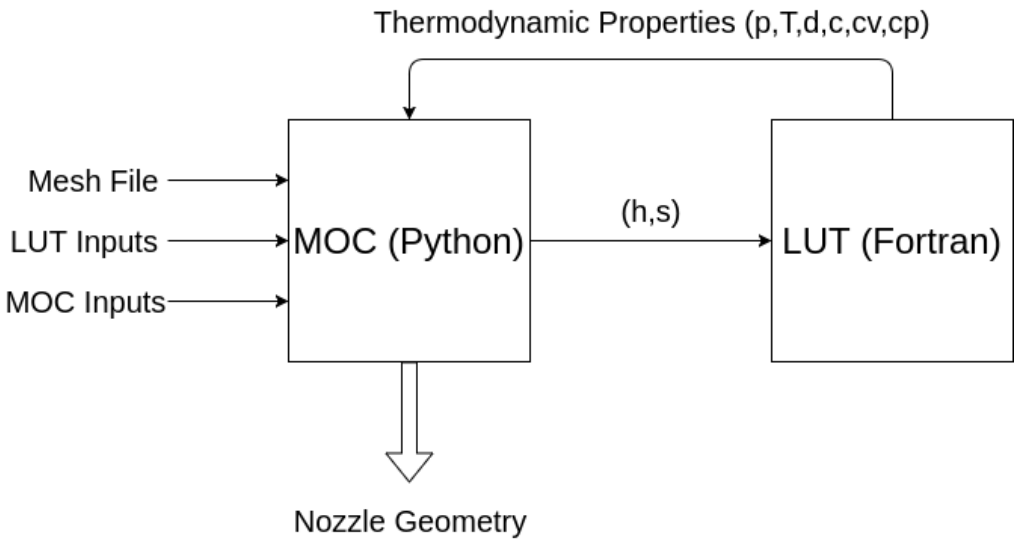



Figure 5.4: Data Flow between LUT tool and MOC tool

### 5.2.3. Supersonic nozzle design test case and results

The MOC tool is used to design the diverging section from the throat to the exit of the supersonic nozzle. The starting inlet conditions and the properties of the working are listed in table 5.3 . The input files for both LUT tool and MOC tool are included in Appendix A. For the design, $Toluene$ is chosen as the working fluid and the design Mach number is 2.0. For the analysis of the supersonic nozzle design and the computational time, the design test case is run for both the LUT method and *FluidProp* (through the LUT tool).

| Parameter | Symbol | Value | Units |
|---|---|---|---|
| Total enthalpy | $h_{t0}$ | 676.019 | J/kg |
| Total temperature | $T_{t0}$ | 580 | K |
| Entropy | $s_{t0}$ | 0.9 | kJ/K |
| Total pressure | $p_{t0}$ | 20.4e5 | Pa |
| Design Mach Number | $M_{des}$ | 2.0 | – |
| Critical pressure | $T_c$ | 591.79 | K |
| Critical temperature | $p_c$ | 2.09x10$^6$ | Pa |
| Compressibility factor | $z$ | 0.7 | - |

Table 5.3: Fluid Properties and Inlet conditions for nozzle design

#### Thermodynamic grid

A thermodynamic grid based on the pre-existing $(h,s)$ values is generated. These pre-existing values are obtained from nozzle design data using *RefProp*. The grid size is varied between 3075 nodes to 15810 nodes with mesh refinement downstream of the isentropic expansion process. The isentropic expansion process on the thermodynamic grid generated on (h,s) domain can be seen in figure 5.5.

#### Geometry of the supersonic nozzle

It can be seen from figure 5.6 that the discontinuity in the nozzle geometry close to the exit of the nozzle for the coarse mesh increases. An error in the exit area of the nozzle can also be noted with respect to *FluidProp*. These differences in design can be explained by higher errors in thermodynamic properties due to interpolation on the coarse mesh. The MOC requires the mass flow rate at the all design iterations to be nearly constant, however for coarse meshes, this design condition is not met and the mass flow rate varies.

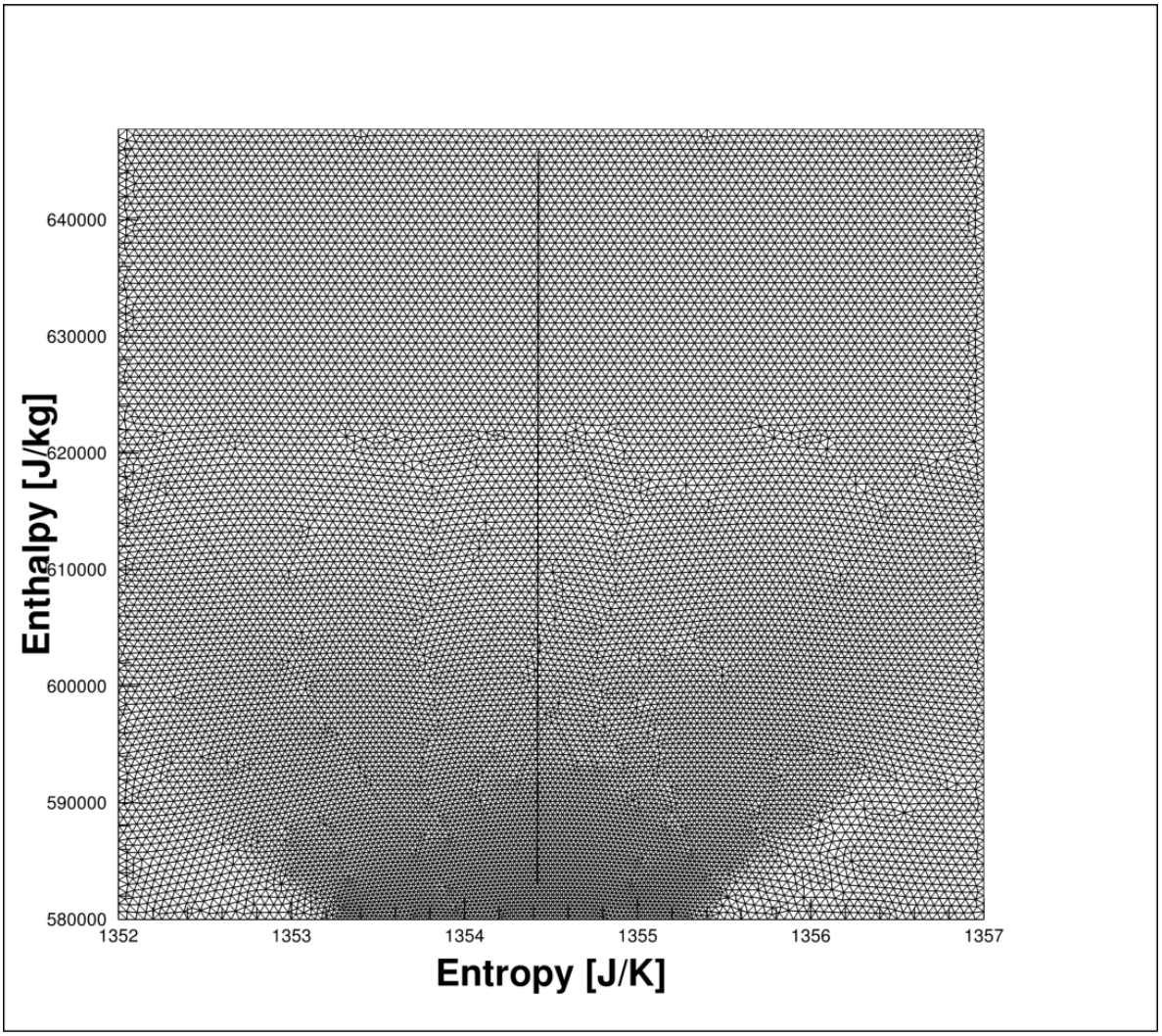
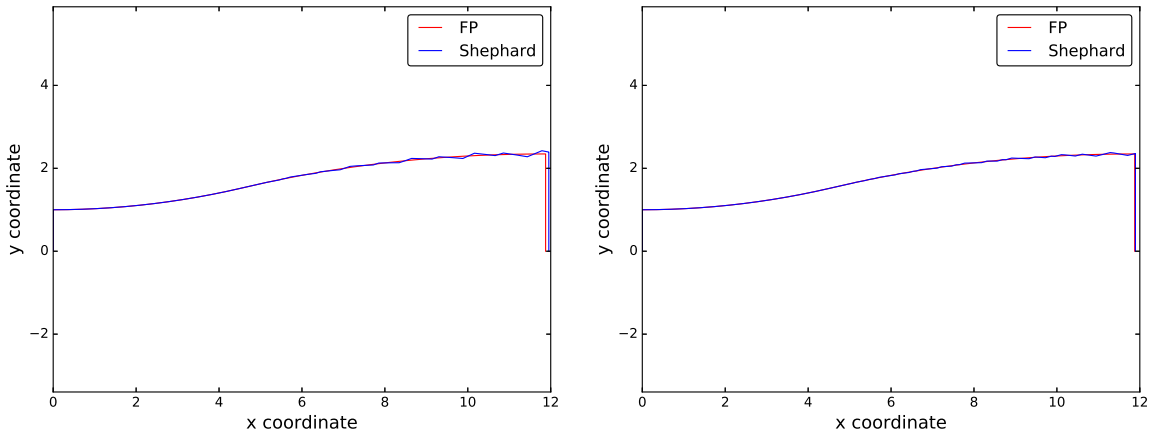
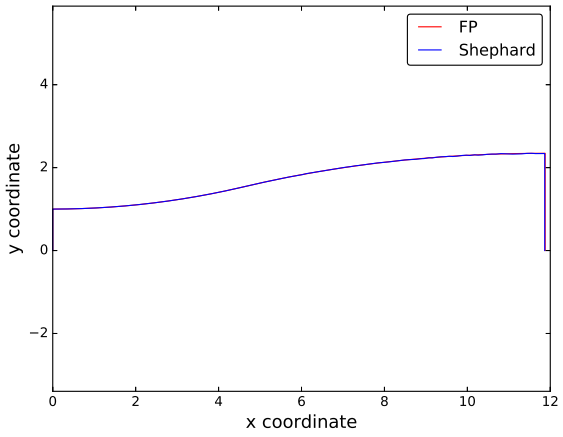



Figure 5.5: $h-s$ thermodynamic domain for the expansion through the supersonic nozzle

As the mesh becomes finer, the errors in thermodynamic properties received by MOC through LUT tool decreases, thereby reducing the variation in the mass flow rate at all design iterations. With mesh refinement downstream in the expansion process (close to the nozzle exit), the nozzle geometry improves significantly, as can be seen from figure 5.6c.




(a) Nozzle geometry designed by mesh nodes $n = 3075$ (b) Nozzle geometry designed by mesh nodes $n = 5435$




(c) Nozzle geometry designed by mesh nodes $n = 15810$

Figure 5.6: Variations in design of the supersonic nozzle geometry using LUT method (Shephard's Interpolation) for feeding thermodynamic properties to the nozzle design tool MOC

An increase in accuracy of the nozzle geometry is noted as the number of mesh elements increases. This is evident from figure 5.7 , which shows the variation of area ratio of the nozzle with the increasing number of grid nodes. The area ratio is defined as the ratio of area at the exit of the nozzle to the area at its throat. With increasing number of nodes and grid refinement towards the exit of the nozzle, the accuracy of the nozzle geometry *FluidProp* increases. The increase in the accuracies of the thermodynamic properties translates into an increase in the accuracy of the geometry.

### Computational cost

The computational cost of the overall design process can be divided into three parts:

1. Time for for pre-processing: This part involves the time required by the design process to read the data from the LUT input file and the thermodynamic tables for each iteration. Figure 5.8 shows that the pre-processing time is the same for both LUT algorithm and *FluidProp*.

2. Time for thermodynamic property calculation: For every iteration, the MOC tool passes the query vector to the LUT algorithm to calculate the required thermodynamic properties. Figure 5.8 shows that for

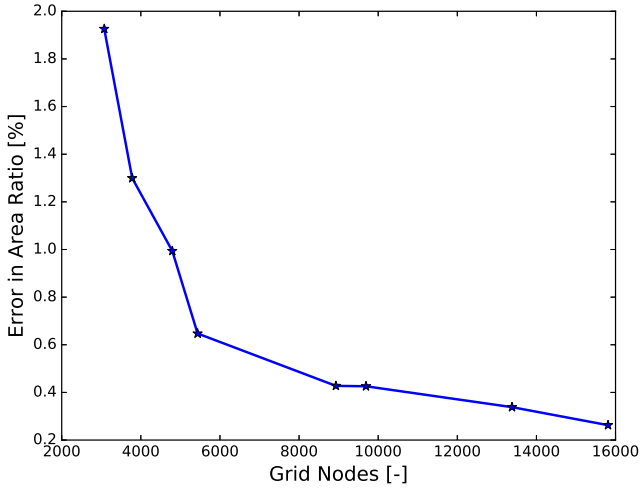



Figure 5.7: Percentage error in exit to throat area ratio for the nozzle design

actual property calculation, LUT is 300 times faster than *FluidProp*.

3. Time for MOC calculation: The time required by MOC tool for its internal calculations is the same for both LUT and *FluidProp*. This can also be seen from figure 5.8.

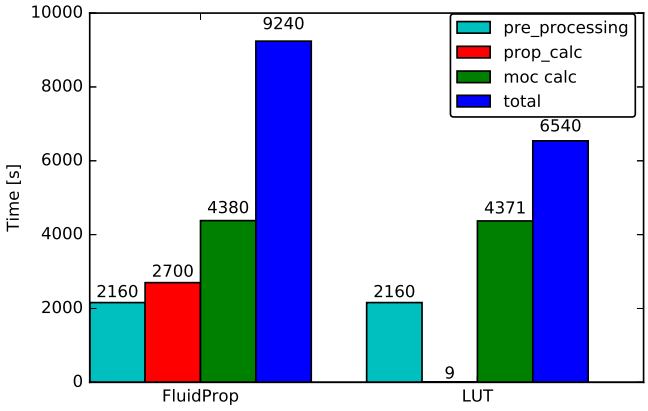



Figure 5.8: Time comparison between *FluidProp* and LUT for individual phases of the nozzle design using MOC.

The overall time ratio for nozzle design using LUT and *Fluidprop* is approximately 1.4. The time ratio reduces further as the number of grid elements increase, because of increasing number of data points. This leads to an increase in the pre processing time for LUT operations, that is, reading of mesh file and thermodynamic tables. This overall ratio can further be improved by eliminating the pre processing stage for each iteration, as a part of future work.

# 6

# Conclusions and Future Recommendations

## 6.1. Conclusions

In order to reduce computational cost for thermodynamic property calculation, a Look Up Table method is implemented. The presented LUT method makes use of Shephard's interpolation and bi-cubic bi-variate interpolation to compute the properties of interest. These interpolation methods work in cohesion with the binary search algorithm, namely the k-d tree algorithm. The accuracy and the computational time of the properties obtained by the LUT method is computed with respect to the properties obtained using *FluidProp*, an external thermo-physical library. To test the applicability of the method, an isentropic expansion process under non ideal flow conditions is considered. As a second application, the LUT tool is integrated with an in house MOC tool for supersonic nozzle design. The nozzle geometry obtained using LUT and *FluidProp* are compared for accuracy in design and computational time.

With respect to the presented work, the following conclusions can be drawn:

- The research presents a successful implementation of an LUT method based on integration of array indexing operations on existing thermodynamic tables (obtained as a part of pre-processing), interpolation methods (Shephard's interpolation and Bi-cubic interpolation) and the k-d tree search algorithm.
- The LUT method is capable of working with a variety of thermodynamic input pairs. Pairs such as $(v, s)$, $(p, T)$, $(h, s)$ and $(p, s)$ are successfully tested for accuracy and computational time. The applicability of pairs $(p, s)$ and $(p, T)$ is tested with the isentropic expansion process whereas the use of $(h, s)$ is tested with the supersonic nozzle design application through MOC.
- The LUT method proves to be computationally efficient for the thermodynamic property calculation. For 10000 points tested, the LUT method with both Shephard's interpolation and bi-cubic bivariate interpolation proves to be faster (for both MDM and CO2) than direct property calculation using *FluidProp*, while maintaining high accuracies in thermodynamic properties at the same time. This gain in computational efficiency can be attributed to the replacement of direct property calculation using complex EoS by simple array indexing operations on existing thermodynamic tables for the respective fluids.
- The LUT method is used successfully to analyse an isentropic gas flow through a single inlet and outlet control volume for subsonic flow conditions. The direct comparison of the results obtained using LUT with *FluidProp* shows that the LUT method is approximately 5.7 times faster and has a minute inaccuracy with respect to *FluidProp*. This process also validates the accurate application of the bisection method within the LUT tool, for solving the energy equation in implicit form to obtain the static pressures across the discretized control volume.
- The coupling of the LUT tool with the supersonic nozzle design tool MOC presents the second working application of the LUT method. The LUT tool is approximately 1.4 times faster than *FluidProp* for the overall nozzle design. For the computation of thermodynamic properties of all query points alone (excluding pre-processing and MOC operation), the LUT method is 300 times faster than *FluidProp* on an average for various mesh sizes tested.

Overall, the LUT method presents a successful alternative for thermodynamic property calculation with a high computational efficiency and accuracy in properties. The tool also shows successful applicability for an isentropic expansion through a control and the geometry design for a supersonic nozzle (coupled with MOC).

## 6.2. Future Recommendations

Based on the outcomes, results and conclusions of the thesis project, the following recommendations for future work can be made:

- Thermodynamic domains other than $(T, s)$ and $(h, s)$ can be built used for the analysis of the LUT method. These domains can then be tested with the implemented interpolation methods and search algorithm to test their functionality and compare the results obtained with the results in chapter 4 and chapter 5.
- A study can be carried out on whether more complex methods for scattered data interpolation, such as Radial Basis Functions [10] or Krigging [10] can be implemented in addition to bi-cubic interpolation and shephard's interpolation scheme to further enhance the accuracy of the LUT method.
- Implementing bi-cubic bi-variate interpolation for MOC tool can further reduce the computational errors in the nozzle design process. Although this interpolation method is already implemented in the LUT tool, it does not work currently for the nozzle design process through MOC.

# A

## A.1. MOC Input File

The contents of the MOC tool input file used for the analysis in chapter 5 is as shown below:

```
# PLANAR||delta=0 or AXISSYM||delta=1

NozzleType PLANAR

# 'PRFT' = Perfect Gas, EoS = Equation of State, EoS_TAB = Using Table, LUT = Look Up Table

GAS_EQU LUT

FLDNAME=Toluene
THROAT_PROP = FILE

#EoS: Solve - Numerical ise: NULL
VrhoGuess = 0.03
TGuess = 580
PGuess = 20e5

# RealGas Table #'Create' 'Calc'
Table_Oper Calc
LLim = 200
ULim = 700
dV = 10


# REAL GAS PROP - SI Units (Toluene) (ref: Wiki)
Tc = 591.79                     # K
Pc = 2.09e6                     # Pa
Vc = 0.000316                   # m^3/mol
M = 0.09214                     # kg/mol
Ho = 676.019                    # J/Kg
so = 0.9                        # J/kg.K
gamma = 1.055                   # cnst
To = 580                        # K
Po = 20.4e5                     # Pa
R = 8.314                       # mol/Kg/K

# Input: Throat Design
```

```
rho_t 20
y_t 1
rho_d 20
n 20

# Accuracy: Accuracy Required

#Corr_n 1000
tolerance_v 1e-9
tolerance_x 1e-9
#Corr_n_inv 100
dtau 0.1
#Reflex:

n_ref = 50
# Design Output: Output Mach Number

Noz_Design_Mach 2.0

# WriteData: Writing Variables

File_Name_NProp Nozzle_prop.out
VAR x y u v M rho SoS T P

File_Name_NCods Nozzle_coords.out
```

## A.2. LUT Input File Data

The contents of the LUT input file used for the analysis in chapter 5 is as shown below:

```
fluid                       : Toluene
unit                        : SI
database                    : Refprop
search algorithm type       : kdtree
interpolation method        : shephard's interpolation
pair query vector           : (h,s)
query vector                : input from MOC directly
```

# Bibliography


[1] N. Anand. *Supersonic Turbine Design using Method of Characteristics*. Msc. thesis, Department of Process and Energy, Faculty of Mechanical, Maritime and Materials, Delft University of Technology, September 2016.

[2] Ian H. Bell, Jorrit Wronski, Sylvain Quoilin, and Vincent Lemort. Pure and pseudo-pure fluid thermophysical property evaluation and the open-source thermophysical property library coolprop. *Industrial & Engineering Chemistry Research*, 2014.

[3] J. L. Bentley. Multidimensional divide and conquer. *Communications of the ACM*, 23(4):214–229, 1980.

[4] Yunus A. Cengel and Michael A. Boles. *Thermodynamics: An Engineering Approach, Eighth Edition*. McGraw Hill - Education, 2015.

[5] P Colonna and TP Van der Stelt. Fluidprop: a program for the estimation of thermo physical properties of fluids. *Energy Technology Section, Delft University of Technology, Delft, The Netherlands, http://www. FluidProp. com*, 2004.

[6] Piero Colonna, John Harinck, Stefano Rebay, and Alberto Guardone. Real-gas effects in organic rankine cycle turbine nozzles. *Journal of Propulsion and Power*, 24(2):282–294, 2008.

[7] Piero Colonna Enrico Rinaldi, Rene Pecnik. Exact jacobians for implicit navier–stokes simulations of equilibrium real gas flows. *Journal of Computational Physics*, 459-477:459–477, 2014.

[8] S. Vitale A.J. Head M. Pini P. Colonna G. Gori, A. Guardone. Non-ideal compressible fluid dynamics simulation with su2: Numerical assessment of nozzle and blade flows for organic rankine cycle applications. October 12-14, 2015.

[9] John D Anderson Jr. *Fundamentals of Thermodynamics, Fifth Edition*. McGraw Hill Education Private Limited, 2014.

[10] J.P. Lewis Ken Anjo and Frederic Pighin. *Scattered Data Interpolation for Computer Graphics*, 1st volume. *SIGGRAPH 2014*, 2014.

[11] D.E. Maiden. *Thermodynamic Interpolation*, nuclear explosives development conference. 1998.

[12] Hill P.G. Miyagawa K. A tabulated taylor series expansion method for fast calculation of steam properties. *Journal of Engineering for Gas Turbines and Power, ASME*, 119:485–491, 1997.

[13] Andrew W. Moore. *Efficient memory based learning for robot control*. Phd. dissertation, University of Cambridge, 1991.

[14] P. K. Nag. *Engineering Thermodynamics, Fourth Edition*. Tata McGraw Hill Education private Limited, 2010.

[15] M Pini, A Spinelli, G Persico, and S Rebay. Consistent look-up table interpolation method for real-gas flow simulations. *Computers & Fluids*, 107:178–188, 2015.

[16] Enrico Rinaldi, Rene Pecnik, and Piero Colonna. Exact jacobians for implicit navier–stokes simulations of equilibrium real gas flows. *Journal of Computational Physics*, 270:459–477, 2014.

[17] Donald Shephard. A two-dimensional interpolation function for irregularly-spaced data. *Proceedings of the 23rd ACM national conference*, pages 517–524, 1968.

[18] R. Span and W. Wagner. Equations of state for technical applications. i. simultaneously optimized functional forms for nonpolar and polar fluids. *International Journal of Thermophysics*, 24(1):1–39, Jan 2003.

[19] R. Span, W. Wagner, E.W. Lemmon, and R.T. Jacobsen. Multiparameter equations of state - recent trends and future challenges. *Fluid Phase Equilibria*, 183-184:1–20, 2001.

[20] Koichi Watanabe and R B. Dooley. Guideline on the tabular taylor series expansion (ttse) method for calculation of thermodynamic properties of water and steam applied to iapws-95 as an example. *International Association for the Properties of Water and Steam*, 2013.

[21] P. Colonna W.C. Reynolds. *Thermodynamics Fundamentals and Engineering Applications*. AE1240-I: TU Delft, 2016.