RANS Modelling of a LOX/CH4 Single Injector Rocket Combustion Chamber

BENJAMIN MCNUTT

Academic Supervisor: Dr. Ben Thornber
Industry Supervisor: Dr. Justin Hardi
Industry Supervisor: Jan van Schyndel

A thesis submitted in fulfilment of
the requirements for the degree of
Bachelor of Mechatronic (Space) Engineering with Honours

School of Aerospace, Mechanical and Mechatronic Engineering
The University of Sydney
Australia

13 July 2022

Acknowledgements

Partaking in this thesis has been a highly rewarding and memorable experience. I would like to thank the University of Sydney for facilitating the ESIPS joint venture with DLR. The opportunity to undertake a thesis project in industry, whilst being able to travel overseas and be apart of the DLR team on-site has been truly special. I extend my thanks to all my supervisors, who were super helpful in guiding the thesis through to completion. A huge thank you to my university supervisor Ben Thornber who supported me through the whole process, especially with getting started and finishing up. The previous student, Clara Morris was also extremely helpful in getting me up to speed with the project before I headed off to Germany.

I would also like to thank everyone at DLR, Lampoldshausen. Whilst I was definitely a fish out of water arriving in outback Germany, you all ensured I had a smooth transition and I thoroughly enjoyed my 3 months working with everyone. A massive thanks to Justin Hardi who oversaw the whole process and offered valuable feedback throughout. I would also like to thank Michael Boerner, who lead the student group meetings. A highlight was being able to work with other students from around the world, learn about their projects and exchange feedback. Finally, none of this would of been possible without my supervisor Jan Van Schyndel. Thank you for the consistent technical help, valuable feedback and general assistance with getting myself setup at DLR. I really appreciate the help you gave me and I've learnt so much under your guidance. I also enjoyed the outings around town and the introduction to German beer!

I would also like to thank Beate and Udo Frank for allowing me to stay in their home. You were both super friendly and made me feel right at home in Lampoldshausen. Lastly, I want to acknowledge my support back home. To my friends, family and girlfriend Andrea, your consistent messages of motivation were greatly appreciated. Love you all!

Executive Summary

This thesis presents the Reynolds-Averaged Navier-Stokes (RANS) modelling of "Brenkammer N" (BKN), a LOX/CH_4 single injector rocket combustion chamber operated by the German Aerospace Centre (DLR). Utilising ANSYS Fluent as the computational fluid dynamics (CFD) code, the steady state flow field has been modelled with the aim of improving the simulation fidelity established by a previous student. Additionally, an extensive survey of literature has informed a sequential analysis of different facets of the simulation setup, detailing their effects with comparison to experimental studies.

The influence of differing boundary conditions, combustion model parameters, turbulence model parameters, turbulence models, chemical mechanisms and species transport modelling approaches have all been explored. Chamber wall boundary conditions were found to impact wall heat flux results significantly. A conduction gradient boundary, incorporating wall thickness and material properties improved wall heat transfer results as opposed to using fixed temperature profiles. The PDF-flamelet method was selected as a computationally feasible method of modelling combustion. An analysis of its tabulation parameters was performed to ensure an accurate capture of the chemistry. Minimal impact was seen in the final results when compared to the default setup. The turbulent Schmidt number was lowered from 0.85 to 0.55, with lower values inducing higher wall heat fluxes and lower wall pressures. A more rapid consumption of oxygen and higher axial peak temperatures was also noted as the turbulent Schmidt numbers was lowered. Fluent formulations for resolving the near-wall boundary layer predicted wall pressures closer to experimental when compared to standard wall functions.

The standard k-ϵ model was chosen as the primary method of modelling turbulence. Alternatives including the RNG k-ϵ model and Generalised k-ω (GEKO) model were explored, predicting significantly different flame topologies, with unburnt LOX cores exiting the nozzle. The GRI3.0 and RAMEC chemical mechanisms were compared to schemes developed at

DLR, with minimal difference seen between them. As the simulation setup became for complex, namely the inclusion of individual species transport modelling and realistic mixing laws, the chemical mechanism had a larger impact. The Wilke mixing law was used for viscosity with the Herning/Zipperer mixing law used for thermal conducutivty. The inclusion of these laws over a mass-weighted approach predicted lower wall pressures and higher wall heat fluxes, conforming more closely with experimental measurements.

Whilst a simulation of improved fidelity was achieved, two large discrepancies remain between the numerical and experimental results. Wall heat flux predictions are approximately half of what was recorded experimentally. This is attributed to multiple model simplifications, including the one-dimensional wall conduction gradient and the limitations of the PDF-flamelet model. Additionally, the correct flame length remains unknown. DLR's in-house solver TAU and the alternative turbulence models explored in this work produce flames not contained within the geometry. Additionally, unsteady flame detachment was observed experimentally with the flame extending beyond the optically accessible region. Further experimental and numerical studies are required for a more definitive agreeance in flame length and wall heat flux, an avenue of pursuit for a following ESIPS student.

Contents

List of Figures

List of Tables

Nomenclature

Acronyms

BKN	Brennkammer N
CEA	Chemical Equilibrium and Applications
CFD	Computational Fluid Dynamics
CH_4	Gaseous Methane
DLR	German Aerospace Centre (Deutsches Zentrum für Luft- und Raumfahrt)
DNS	Direct Numerical Simulation
EDC	Eddy Dissipation Concept
EoS	Equation of State
ESA	European Space Agency
EWT	Enhanced Wall Treatment
GEKO	Generalised k-ω
H_2	Gaseous Hydrogen
ISS	Internal Space Station
JL	Jones-Lindstedt
LES	Large Eddy Simulations
LH_2	Liquid Hydrogen
LOX	Liquid Oxygen
NASA	National Aeronautics and Space Administration
NIST	National Institute of Standards and Technology
PDF	Probability Density Function
PR	Peng-Robinson
RANS	Reynolds-averaged Navier-Stokes
RNG	Renormilisation Group
SRK	Soave-Redlich-Kwong

UDF	User Defined Function
UoS	Unit of Study
USYD	University of Sydney
WHS	Work Health and Safety

CHAPTER 1

Introduction

The rapid evolution of space technology over the past century has undoubtedly transformed the world and set a new standard of what human-kind is capable of. It is even more potent now, as the dawn of a new space age emerges with the likes of private space companies reshaping the industry and the next pursuit of crewed missions to the moon and beyond. Whilst the importance of evolving space technology and aiming for new heights can be discussed extensively, innovation does not occur by itself. It requires the constant research and development undertaken by engineers and scientists to maintain the rapid momentum seen today. Arguably the most critical element that has facilitated all space travel to date, is the means of propulsion.

Whilst numerous methods of propulsion have been explored, liquid rocket engines are used the most extensively as the primary means for launch vehicles to reach space. Numerical methods of simulating these engines have become a powerful tool in recent years, yet the combining effects of combustion and highly turbulent flow is still far from properly understood. Without experimental validation, simulation results are not yet sufficiently reliable. Hence, as an ongoing and relevant area of research, accurate numerical models will continue to be a critical avenue of pursuit.

1.1 Project Aim

The aim of this project and thesis is to accurately model a single injector, high pressure combustion chamber. The German Aerospace Centre (DLR) operates an experimental chamber called "Brennkammer N" (BKN), built with an extensive suite of sensors and tools to analyse the combustion dynamics, most notably its large optical access window. In conjunction with DLR, numerical analysis through a Reynolds-averaged Navier-Stokes (RANS) simulation will be undertaken. More specifically, the model will employ liquid oxygen (LOX) as an oxidiser and gaseous methane (CH_4) as a fuel.

Currently a lack of numerical validation test cases exists, largely attributed to the combination of complex phenomena occurring within the chamber. With the liquid oxygen injected cryogenically, steep thermal gradients are present within a highly turbulent reactive flow. Combined with high pressures, non-premixed injection and supercritical effects, numerical simulation of this type of fluid flow is a challenging task. As DLR is a world leader in published data concerning combustion chamber analysis via a large optical access window, an accompanying numerical model is a high priority. Thus the motivation for this project arises, with the ANSYS suite of software selected as the primary tool. A direct comparison to the observed flame topology will be made, in addition to conventional data taken from the sensors inside. DLR's in-house solver TAU will act as an additional source of comparison.

This project is taken as a continuation from a previous student who completed 6 months of work on this topic. A single baseline simulation was developed, a starting point for which more detailed analysis is required. The simulation fidelity is the main focus, with many areas previously left unexplored. With the maintained pursuit of this project, this body of work will add to the rapidly growing area of research surrounding LOX/CH_4 combustion.

1.2 DLR ESIPS Placement

The incredible opportunity to work with DLR as a part of this thesis project is made possible through the Engineering Sydney Industry Placement Scholarship (ESIPS). DLR have sites at 35 different locations, with this placement completed at the Institute of Space Propulsion in Lampoldshausen. Collaboration with The Research Group for Combustion Dynamics and Flows guided this thesis through to completion.

The DLR site in Lampoldshausen is a key partner for the European space industry and has hosted the development and operation of rocket engine test stands for over 60 years. It was setup by Eugen Sänger, a leading rocket scientist in Germany during World War II. After the ban on rocket research lifted in 1955 he founded the Research institute for the Physics of Jet Propulsion Systems, requiring a liquid rocket engine test site [31]. In 1959, in the Hardthäuser forest, the DLR Lampoldhsusen site was was setup and expanded rapidly throughout the early 1960's. Now it contains multiple research institutes and test stands used by organisations all over Europe. Most notably, the development of the European Space Agency's (ESA) successful Arianne rocket, with current testing being conducted on the Vulcain 2.1 and Vinci engine. In conjunction with various research groups, the site welcomes students writing bachelor, masters and PhD thesis' as well as interns and trainee technicians.

(A) ESA LOX/CH_4 demonstration engine test

(B) ESA Vulcain 2 engine static test fire

FIGURE 1.1. Example rocket engine test fires at DLR Lampoldshausen [24]

1.3 Mode of Completion

This thesis was conducted remotely in Sydney, Australia for the first 3 months before being completed on site in Lampoldshausen, Germany. International travel and the surrounding logistics was factored into the project planning and timeline. A mixed completion mode was necessary due to the COVID-19 pandemic, as home-office rules were enforced in Germany during the first 3 months.

1.4 Thesis Structure

This thesis is structured as follows:

Chapter 2 Work Health and Safety (WHS) contains a brief overview of the WHS practises abided by for the duration of the thesis. Considerations for both home-office and on-site modes are discussed.
Chapter 3 Background outlines the relevant background information surrounding space propulsion and liquid rocket engines in particular. Additionally, a detailed survey of literature covering experimental and numerical methods for analysing liquid rocket engines with similar conditions to BKN is presented.
Chapter 4 Case Study 1 covers the course content of the USYD UoS AMME5202 Computational Fluid Dynamics with relevance to the numerical study of BKN. The governing equations, RANS decomposition, closure methods and computational domain are all detailed.
Chapter 5 Case Study 2 covers the course content of the USYD UoS MECH5265 Combustion with relevance to the numerical study of BKN. Combustion kinetics, chemical mechanisms and approaches to embedding combustion within a turbulent flow are all presented.
Chapter 6 Methodology details the research design, software procedures, numerical preprocessing and simulation setup.
Chapter 7 Results presents a detailed discussion of all the collated results with relevance to the thesis objectives, literature and the modelling techniques outlined in the Chapters 4-6.
Chapter 8 Conclusion summarises the completed work and defines further avenues of exploration.

FIGURE 1.2. DLR Lampoldshausen front office and museum [26]

CHAPTER 2

WHS

This chapter presents the WHS practises followed for the project duration. Sec.2.1 presents the considerations for the remote work conducted in Sydney. Sec.2.2-2.5 proceeds to outline the considerations surrounding international travel and on-site work in Lampoldshausen, Germany.

2.1 Remote Work

2.1.1 COVID-19

The beginning of the project took place at the end of the Omicron COVID-19 variant outbreak in Sydney. Whilst no lockdown or stay at home measures were enforced, a restraint from public gatherings and regular mask wearing was practised. The authors housemate contracted the virus, requiring a week of isolation. A booster vaccine was also sought to increase protection and meet the requirements for travelling to Germany.

2.1.2 Workstation Setup

In order to operate effectively from home, a suitable home office conducive to long hours of work was setup. Investments were made into a large second monitor for increased productivity and a reduction in eye strain and a sit-stand desk to aid in back health. In addition, regular breaks were taken which often involved exercise and stretching.

2.2 International Travel

The logistics surrounding travel to Germany were complex due to the heightened level of COVID-19 transmission in Germany. Approval was sought from the Vice-Chancellor after completing a risk assessment and consulting the smart traveller site [40]. Aside from the pandemic, all other risks were considered with backup measures put in place such as:

- Complete backup of laptop in case of theft or damage
- Flexible flight tickets due to unexpected changes in plan
- Multiple copies of international vaccine certificates, bank cards, insurance forms and other important documentation
- Secure travel wallets, locked luggage and insurance on valuables
- Backup funds and sources of financial help

2.3 Ukraine Crisis

A month before travelling, Russia declared its special military operation in Ukraine. Despite Lampoldshausen being 2000km from the warzone, it was still a consideration given the overall geographical proximity, Russia's volatility and Germany's political involvement. In preparation for the worst, escape routes via train to family friends in Austria and family in the UK were devised.

2.4 DLR Security and Safety Briefing

In order to adhere to all WHS practises in place at the DLR site, an in depth briefing was given 3 days after arriving. The following was covered:

- Site map overview, restricted areas, evacuation assembly points, first aid stations etc.
- Alarm sounds, warnings and light colour definitions
- Emergency contact numbers, including the internal dedicated response unit
- Accident and hazard reporting

- COVID-19 measures, mask wearing was mandatory inside for the duration of the project
- Sensitive data handling, no photos were allowed apart from in the museum

The completion of the briefing warranted a green pass, allowing unaccompanied movement around the site. A brochure with a summary of what was covered was also supplied in case information gets forgotten.

2.5 Rocket Engine Tests

Routine rocket engine tests were conducted at numerous test benches around the DLR Lampoldhsausen site. Viewings from inside the control rooms were occasionally permitted when accompanied by authorised personnel. Outside viewing was also permitted from a certain distance. A fire truck was always present before each test, with a countdown sounded around the entire site as a general warning. Earmuffs were worn for ear protection.

CHAPTER 3

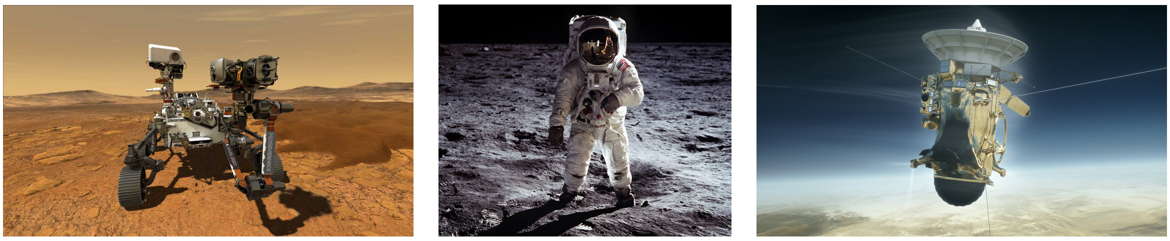
Background

This chapter contains the background information surrounding liquid rocket engines, their design and a comprehensive literature review relevant to the CFD simulation of BKN. Section 3.1 introduces space propulsion and the modern techniques used today. Section 3.2 details the liquid rocket engine, its propellants and an analysis of its design. Section 3.3 discusses BKN's physical design and characteristics in relation to the modelling conducted in this thesis. Finally, Section 3.4 discusses the research into the experimental and numerical methods used to investigate liquid rocket engines. The key findings are the summarised in Section 3.5.

3.1 Space Propulsion

The evolution of propulsion technology underpins the modern space industry. The earliest legend of rocket principles dates back to 400 B.C, when a Greek named Archytas entertained citizens with a wooden bird propelled by escaping steam [9]. Through the centuries since, many more examples based upon the action-reaction principle can be noted, before it was formally defined as a scientific law by Sir Isaac Newton in the 17th century. The foundations for modern rocketry were set, with an understanding of physical motion off which future designs could be based.

As early as 1945, the exploration into the application of cryogenic liquid propellants for space propulsion purposes was underway [16]. Its rapid development and application has since shifted the global landscape completely. The F-1 engine, the backbone of the Saturn-V rocket which launched the Apollo astronauts to the moon re-defined the capability of human-kind. Today, the deployment of thousands of earth orbiting satellites, the conducting of scientific research throughout and beyond the solar system and crewed missions to the international space station (ISS) are routine. Assisting in the furthered understanding, optimisation and improvement of this technology ultimately provides the motivation for this research.



(A) Mars Rover Perseverance [65] (B) Neil Armstrong [103] (C) Cassini [11]

FIGURE 3.1. Examples of modern space technology and practise

Whilst this thesis pertains to liquid bi-propellant combustion, other forms of propulsive techniques exist, namely electrical and nuclear thermal. Electrical propulsion systems use an external electrical power source to accelerate gases, generating an exhaust stream that induces thrust [13]. The main benefits of this kind of propulsion include fuel efficiency, longer operation times, reduced tank sizes and cost. Conversely, the thrust currently achievable is minuscule, generally in the millinewton range. This makes it suitable for deep space and manoeuvres that don't require rapid acceleration, including station keeping, attitude control and orbit transfers [23].

Nuclear thermal propulsion offers an alternate option to electrical methods as it shares similar strengths and weaknesses, albeit with higher possible thrust forces. Currently, the space applications of nuclear technology are generating heat and electrical power and its adoption as a means of propulsion is not expected in the near future [72]. Both electrical and nuclear thermal methods exude a dramatic improvement in specific impulse I_s when compared to chemical propulsion. As a key performance evaluator, it is defined as the ratio of total impulse per unit weight, seen in Eq.3.1. It can easily interpreted as the rocket efficiency, making it a high priority in engine design.

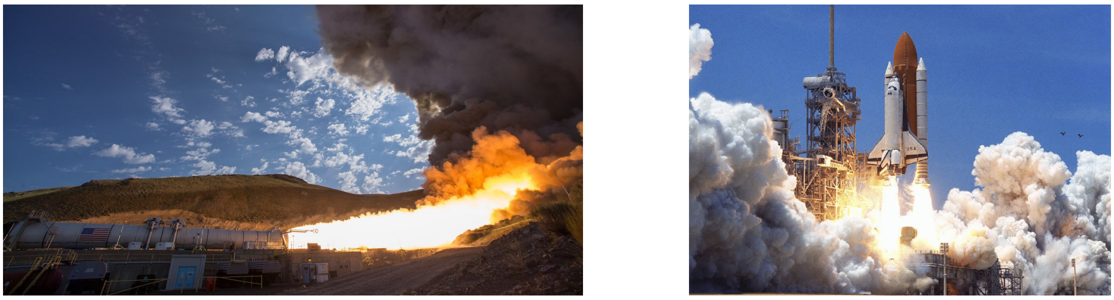
$$I_s = \frac{I_t}{w} \quad \text{where} \quad I_t = \int_0^t F(t)dt \quad \text{and} \quad w = g_0 \int_0^t \dot{m}(t)dt \tag{3.1}$$

3.1.1 Chemical Propulsion

Chemical propulsion is the only method that generates the thrust capable of launching a spacecraft from the earth's surface. Its primary mechanism is an exothermic reaction between a fuel and oxidiser that releases energy, accelerating hot gases and subsequently forming a high velocity exhaust stream. Chemical propulsion typically utilises a combustion reaction and can be defined by the propellant phase and type.

3.1.1.1 Solid Propellant

The simplest means of producing thrust via combustion is the use of a solid cast material that contains both fuel and an oxidiser. Examples include a mixture of nitrocellulose and nitroglycerine forming a "double-base-propellant", or crystals of an oxidizer such as ammonium perchlorate, dispersed in the matrix of a hydrocarbon fuel polymer to form a "composite propellant" [102]. An ignition source is required to begin the reaction, usually through pyrotechnics or electrical heating. The rate at which the solid fuel recedes is a critical design factor as once the reaction begins it will burn to extinction. When compared to liquid systems, solid rocket motors have a lower I_s. Notably, solid rocket motors are being used in NASA's upcoming space launch system (SLS), derived from the space shuttle boosters [42].



(A) SLS solid rocket booster test fire [34]

(B) Space shuttle liftoff [10]

FIGURE 3.2. Examples of solid rocket booster engines

3.1.1.2 Liquid Propellant

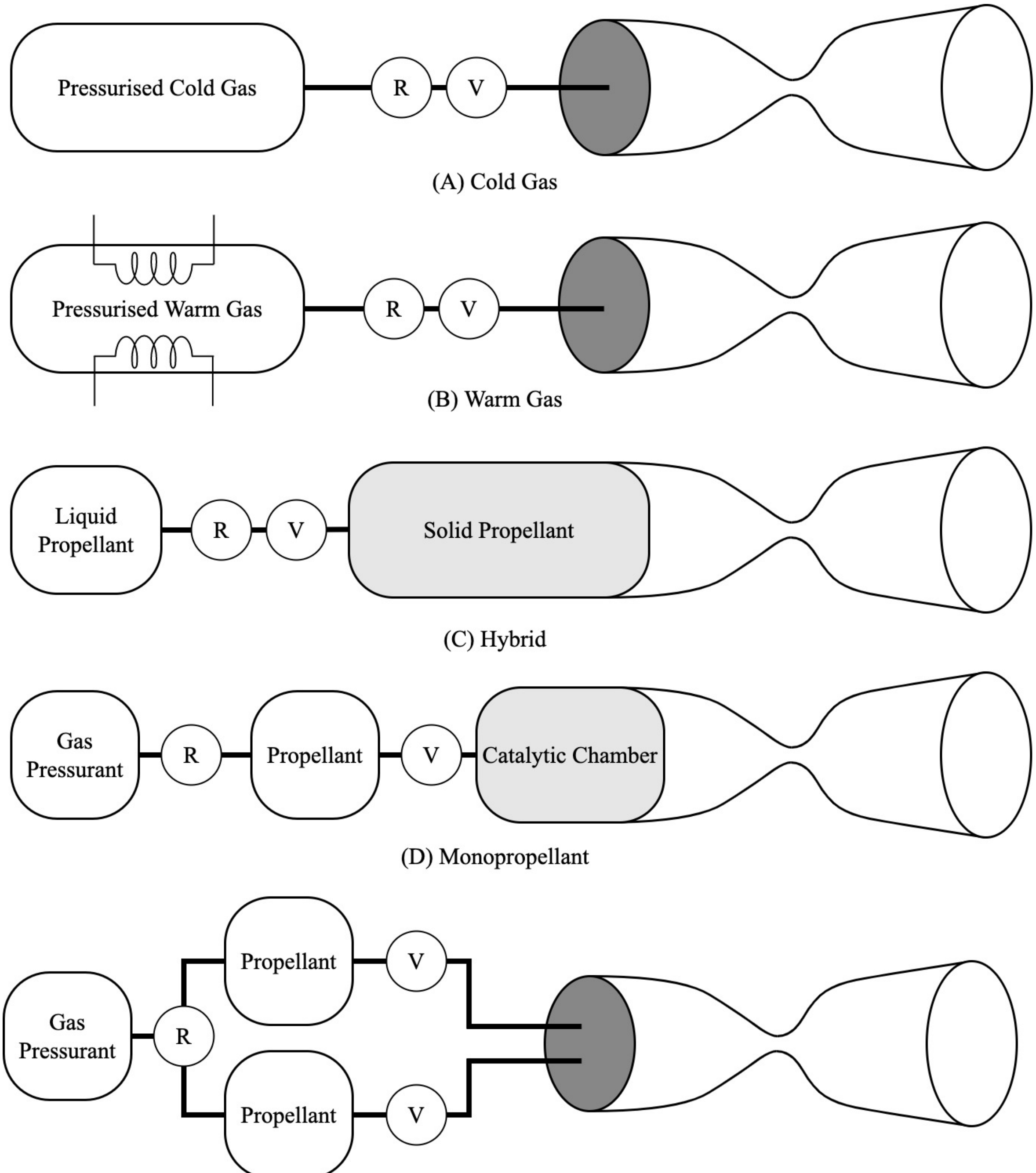
The use of liquid propellants in a chemical propulsion system results in a higher specific impulse and the ability to throttle the engine [53]. This makes them more popular for launch vehicles, however the engine design is far more complex, especially in bipropellant configurations. Figure 3.3 illustrates the various types of liquid rocket engines, summarised as follows:

Cold and Warm Gas - Whilst not explicitly a "chemical" system due to their non-reactive flows, cold and warm gas thrusters form the most basic form of liquid propulsion. Simply put, a working gas is expanded through a converging-diverging nozzle. Warm gas thrusters heat the propellant before expansion to increase the internal energy, and thus the specific impulse. This system is limited by the achievable storage pressure and thrust decays, however is commonly used for attitude control due to their miniaturisation capability[53].

Hybrid - Hybrid engines employ a liquid oxidiser and a solid fuel, mirroring the natural combustion processes seen in nature, such as fires. The hybrid system does address the restarting inability seen in solid rocket engines, however it does not possess the specific impulse capability of liquid bi-propellant propulsion [4].

Monopropellant - As the name implies, a single propellant is used to generate thrust, transforming itself into a hot gas through decomposition [5]. Seen in Fig.3.3, the propellant is pressure fed into the combustion chamber, which is lined with a catalytic material. On contact, the propellants decomposition activation energy is lowered, triggering the process to begin. As a result, the released heat allows a self sustaining process of decomposition to continue [83]. The constituents then accelerate and escape through the nozzle. The choice of fuel generally governs the design of whole chamber, with the trade-off being between toxicity, storability, performance and mass [53]. These systems are most commonly seen on spacecraft, used for velocity and attitude control.

Bipropellant - A liquid bipropellant engine is the most complex yet powerful form of chemical rocket propulsion. A fuel and oxidiser are fed into a chamber that promotes mixing and combustion. Subsequently a high-pressure, high-temperature gaseous mixture, expanded through a converging-diverging nozzle produces a high velocity jet stream inducing thrust [53]. Fig.3.3 details a pressure fed system, however turbo-pumps are commonly used for launch vehicles. Through its use over time, numerous fuel and oxidiser combinations have been selected, often requiring extensive cooling and large tanks. Additionally, extreme temperatures and long operation times necessitate chamber cooling. Overall, the design is far more challenging and labour intensive in comparison to previously discussed propulsive methods [86].



(A) Cold Gas

(B) Warm Gas

(C) Hybrid

(D) Monopropellant

(E) Bipropellant

FIGURE 3.3. Configurations of different liquid propulsion methods

3.2 Liquid Bipropellant Rocket Engine Design

There are many intricate subsystems and components that compose a liquid bipropellant rocket engine. At its core, the sole purpose is to produce a high velocity exhaust stream that generates thrust via the combustion of a separate fuel and oxidiser, propelling the entire vehicle in the direction opposite to the nozzle outflow. The mechanisms that drive this can be divided into 3 major categories, discussed below.

3.2.1 Propellant Type

The choice of propellants largely dictates the design of the engine and the overall performance. Propellants can be hypergolic, which self ignite on contact and remain in their liquid state at normal temperatures. These propellants tend to be highly toxic and corrosive, requiring extreme handling care. Additionally, performance is lower when compared to cryogenic propellants, the type modelled in this thesis [47].

Cryogenics can be defined as the production and application of low-temperature phenomena [17]. Additionally, the boundary considered to be "low temperature" is 120 K, as it encapsulates all the boiling points of the main atmospheric gases, as well as methane [57]. Hence, in order to store propellants in their liquid forms, temperatures in this cryogenic range are required. When stored in this state, higher thrust, specific impulse and smaller tanks are achievable [47]. Liquid oxygen is utilised as the oxidiser which has a boiling point of 90 K, however multiple fuels have been adopted over time. The most common fuels are liquid hydrogen (LH_2) and RP-1, a highly refined kerosene. A notable example of a LH_2 engine is ESA's Vulcain 5 whereas the famous F-1 engine onboard the Saturn V used RP-1[99, 29]. A third type of fuel that is being heavily tested and researched today is methane (CH_4). A 3 way comparison between these fuel types is important as it reveals the shared benefits of using CH_4, giving it its nickname the "goldilocks" fuel. RP-1 excels in its high boiling point, storability and extremely high density, yet hydrogen exudes the highest performance of any fuel ever tested [45]. Summarised in Tab.3.1, CH_4 falls between the two, with a high specific impulse, yet a density and oxidiser to fuel ratio that keeps the storage tanks small and relatively similar in size. CH_4 is also more abundant and considered "greener" when compared to RP-1, which can produce soot, a by-product not captured in the Tab.3.1. This allows it be used as for regenerative cooling without the risk of cracking and deposit[98]. Finally, when compared to H_2, the considerable cooling of a larger tank is not required. The balance between RP-1 and H_2 tradeoffs has made CH_4 an attractive choice.

TABLE 3.1. Comparison of average fuel characteristics in their liquid state

Parameter	**RP-1**	**CH4**	**H2**
Density (g/L)	813	422	70
Oxidiser:Fuel Ratio	2.7:1	4:1	8:1
Specific Impulse (s)	370	459	532
Boiling Point (K)	490	111	20
Byproducts	CO2 & H20	CO2 & H20	H20

The DLR test chamber "BKN" has been recently tested with CH_4 and will be modelled in this work. Another notable mention is the Starship Raptor engine currently under development by SpaceX, a rocket designed for crewed interplanetary travel. CH_4 is manufacturable on mars, hence the added interest [27]. The exploration of CH_4 as a fuel with LOX is made even more pertinent, as no rocket to date has successfully flown with this propellant combination.

3.2.2 Feed System

The rocket engine feed system is designed to transport the propellants from their storage tanks to the thrust chamber at the correct pressure and mass flow rate [95]. The main components include the propellant tanks, feed lines, valves and pressurisation devices. The differentiation between whether the system is pressure fed or pump fed defines how the propellants are delivered to the injectors. Typically, high performance engines will use a pump fed mechanism, whereas the pressure feed design is far simpler making it useful for smaller auxiliary engines [8]. Subsequently, performance for a pressure fed system is optimal when chamber pressure and total impulse is low, whereas a pump design benefits high impulse and high chamber pressures [95].

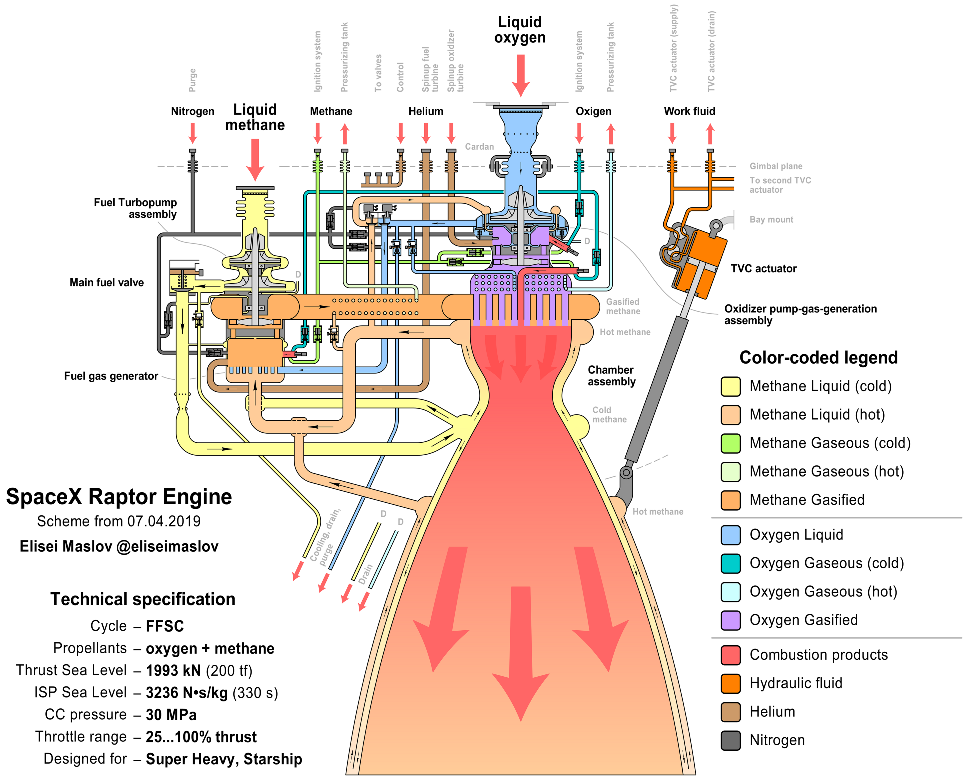
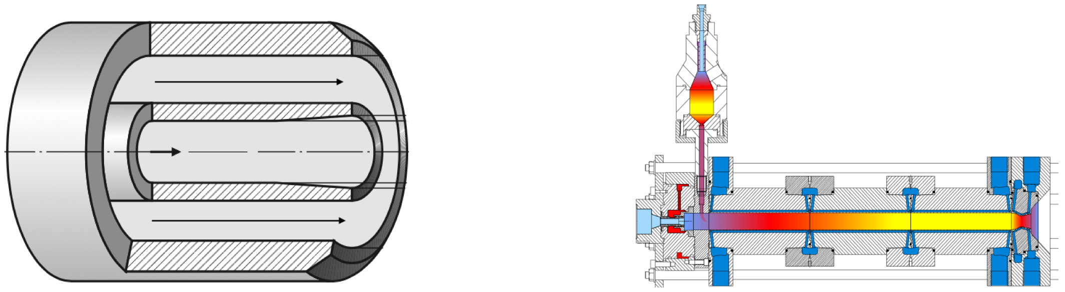


FIGURE 3.4. SpaceX LOX/CH_4 raptor engine design [69]

3.2.3 Thrust Chamber

The principle components of the thrust chamber include the injectors, an ignition device, the combustion chamber and nozzle [47]. The injectors play an important role, ensuring the propellants are fed into the combustion chamber at the proper mixture ratio, pressure, velocity and spray pattern to ensure stable and sustainable combustion. Many designs exist, including the "showerhead" or the "unlike doublet", however the design relevant to this work and propellant combination is the coaxial injector. The oxidiser is injected through the core, whilst the anulus contains the fuel, seen in Fig.3.5a. LOX core recesses, swirlers and tapering can also be added to promote mixing before reaching the combustion chamber[60]. Whilst engines such as the Space Shuttle RS-25 employ 600 coaxial injectors, scaled down research chambers such as BKN have a single injection element.



(A) Single element coaxial injector cross-section [79]

(B) Thrust chamber setup with a torch ignitor [93]

FIGURE 3.5. Examples thrust chamber componentry

The injection streams require an ignition source, unless hypergolic. The igniter releases energy in order to initiate the combustion reaction. Pyrotechnic igniters store chemical energy and release heat once an external stimulus is applied. A torch igniter (augmented spark igniter) generates an electric discharge seen at the region of circulation, triggering the ionisation of the gassified propellants. A flame front is generated, propagating towards the main combustion zone within the chamber [47].

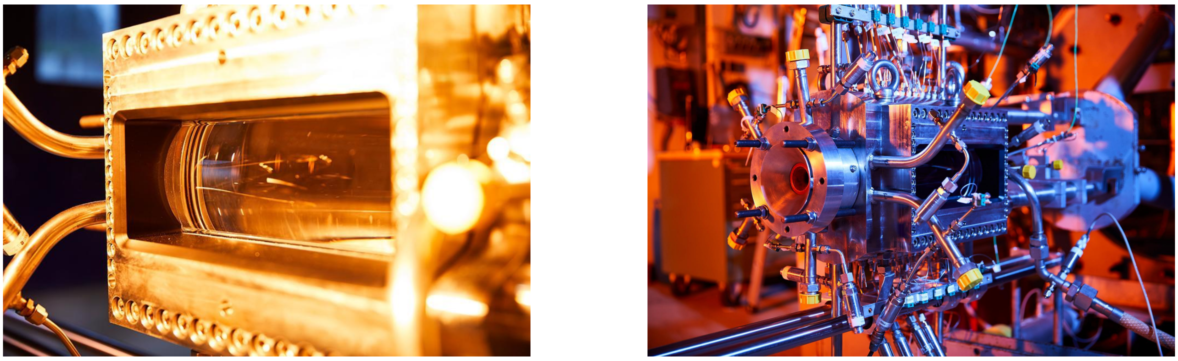
The combustion chamber is tubular in shape and encloses the combustion reaction in a high pressure, high temperature, structurally sound environment. It requires wall cooling, protecting it from the intense heat flux. Combustion temperatures are are typically above 3000K, yet the melting point for most metals is below 2000K [54], hence cooling is paramount for structural integrity. Methods of cooling include[30]:

- **Regenerative cooling** - Utilises the propellants, by running them through channels along the outer wall before being discharged into the combustion chamber, seen in Fig.3.4.
- **Transpiration cooling** - Cooling of the inner wall which is intentionally porous, with a forced cooling fluid passed through it.
- **Film cooling** - A thin layer of cooling fluid is injected along the inside of the chamber wall.
- **Coating** - A layer of low-conductivity material is deposited on the inside of the wall as a thermal barrier.

Finally, a converging-diverging nozzle is employed to accelerate the hot gas products of the combustion reaction out of the chamber at a high velocity. The converging portion of the nozzle is used to increase the fluid to sonic speed, Mach (Ma) number = 1. The throat marks the point of the thinnest diameter where the velocity becomes sonic, followed by the diverging portion of the nozzle which allows the fluid to expand and reach supersonic speeds. For optimum thrust, the pressure at the end of the exit plane of the nozzle should be as close to that of the outside environment as possible [47].

3.3 BKN - DLR's Research Combustion Chamber

As previously mentioned, BKN is a research combustion chamber that has been developed at the DLR, Lampoldshausen site. It is operated by the Research Group for Combustion Dynamics, who are particularly interested in how the chamber design effects the stability of combustion. Additionally, it has been designed as a CFD reference case. It houses a suite of diagnostic tools for sensing capability, with a large optical access window allowing for direct visualisation of the flame behaviour. A window of this size is thought to be the first of its kind [68]. Additionally, it is constructed in a modular fashion, allowing interchangeability with other DLR chambers and engine hardware.



(A) BKN's large optical access window

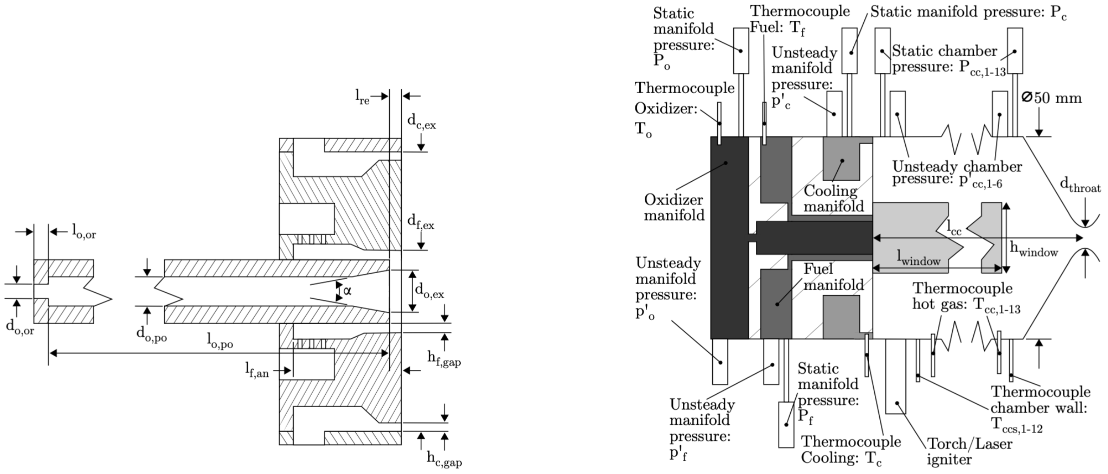
(B) BKN experimental setup

FIGURE 3.6. Images of BKN [25]

BKN is configured as follows:

- **Chamber Dimensions** - The chamber is cylindrical in shape with a 50 mm diameter. It has a length of 359 mm from the injection plane to the nozzle throat, defined as l_{cc} in Fig.3.7b.
- **Injector** - BKN uses a single shear coaxial injector with a recessed and tapered LOX core. The injector is the size of a typical main stage engine injector employing similar mass flow rates. It is designed to be interchangeable, allowing for different types to be investigated.
- **Chamber Cooling** - Film cooling is used, with the fuel injected at room temperature alongside water cooling through external channels.
- **Nozzle** - The converging-diverging nozzle is also interchangeable, allowing the exploration into the effects of design variants. The throat diameter is 14.5 mm.
- **Ignition** - The ignition system utilises a torch ignitor, however it can also be interchanged with a laser system, a new alternative [89].
- **Propellants** - BKN has been tested with LOX/H_2 and LOX/CH_4 propellant combinations [67, 68]. As previously discussed both propellant combinations are of interest for their respective reasons, however the increasing research into LOX/CH_4 constitutes the main reason of its exploration in BKN.
- **Measuring Capability** - BKN has a suite of sensors, notably a range of thermocouples for measuring temperature along the inside of the wall and within the wall. This is useful for temperature profiles in a numerical setup. Pressure sensors are also distributed along the inside of the wall.
- **Optical Access** - The large optical access window spans 255 mm, nearly the entire length of the chamber. Its primary use is for high speed imaging of the flame, notably OH* and CH* radiation seen in the work by Martin *et al.* [67, 68].

An experimental setup such as this provides a focused way of exploring the numerous facets of liquid rocket engine design. The research group for Combustion Dynamics can highlight the influence of physical componentry on the internal combustion processes and flame behaviour. Specific relationships can be drawn, aiding in design modification. Its findings provide test


l_{re}
$d_{c,ex}$
$d_{f,ex}$
$l_{o,or}$
$d_{o,or}$
$d_{o,po}$
$l_{o,po}$
α
$d_{o,ex}$
$h_{f,gap}$
$l_{f,an}$
$h_{c,gap}$
Static manifold pressure: P_o
Thermocouple Fuel: T_f
Static manifold pressure: P_c
Unsteady manifold pressure: p'_c
Static chamber pressure: $P_{cc,1\text{-}13}$
Thermocouple Oxidizer: T_o
Ø50 mm
Unsteady chamber pressure: $p'_{cc,1\text{-}6}$
d_{throat}
Oxidizer manifold
Cooling manifold
l_{cc}
h_{window}
l_{window}
Unsteady manifold pressure: p'_o
Fuel manifold
Thermocouple hot gas: $T_{cc,1\text{-}13}$
Static manifold pressure: P_f
Unsteady manifold pressure: p'_f
Thermocouple Cooling: T_c
Torch/Laser igniter
Thermocouple chamber wall: $T_{ccs,1\text{-}12}$

(A) BKN injection system configuration

(B) BKN experimental configuration

FIGURE 3.7. BKN setup for LOX/CH_4 test fire, taken from [67]

cases for which future experiments can be validated and compared against. Whilst dedicated research chambers such as BKN are useful, other methods of investigating LRE's exist, namely numerical simulations. The combination of both numerical and experimental testing has become a more prevalent approach adopted in research today.

3.4 Numerical Simulation of Liquid Rocket Engines

The development of numerical simulation capability for rocket engines is an increasingly powerful tool that is now essential in their design and testing. Wang categorises the field as multidisciplinary, fusing computation fluid dynamics (CFD), computational heat transfer, computational combustion, computer software design and flow visualisation [100]. Running physical tests is a risky, time consuming and highly expensive process, providing the motivation for simulation capability as a more efficient means of design and testing. It also provides a detailed view of the internal processes which is not possible with experimental setups. Additionally, this pursuit can only enhance the understanding of physical phenomena and thus engine design optimisation. A complete numerical tool to perform the entire design process is yet to exist, with the typical procedure including the derivation and evaluation of existing models, their calculation and a comparison to experimental data [73]. This work will undergo the same process.

3.4.1 Physical Conditions

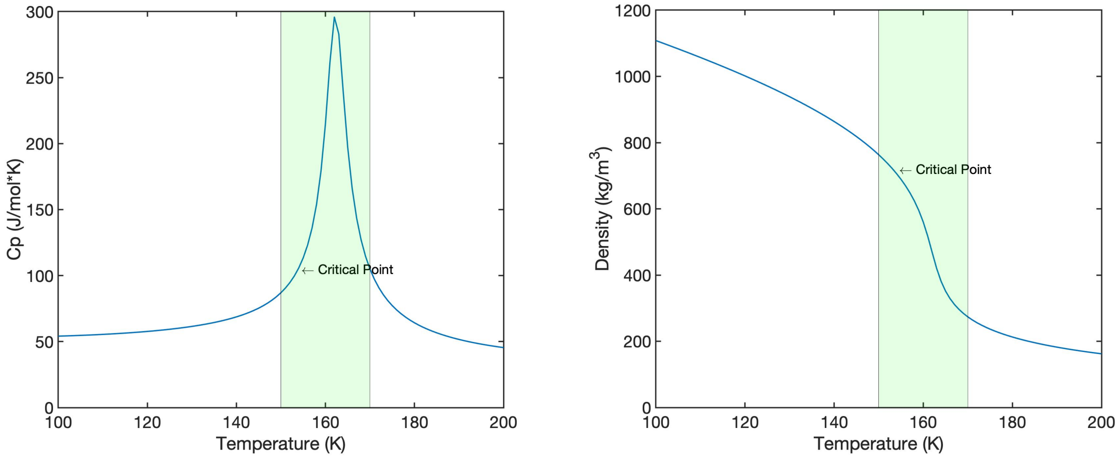
Liquid rocket engines can be characterised by their highly coupled, concurrent physical processes. It is important to detail these conditions and their interplay within the thrust chamber when considering the appropriate numerical approach. In particular, the following phenomena are synonymous with BKN's physical setup, the experimental basis for this thesis.

3.4.1.1 Turbulent Combustion

The governing equations describing viscous flows have been known since 1845, yet with the embedded effects of turbulence and combustion there exists limited ability to predict the flow field with certainty [20]. Whilst the numerical methods for modelling turbulence and chemistry are setup independently, a highly turbulent reactive flow has strong coupling. The heat release that occurs during the reaction induces strong flow accelerations, modifying the kinematic viscosity due to the temperature change. The result is two-fold, with the combustion able to generate more turbulence or damp the existing turbulence. Conversely, turbulence alters the flame that forms during combustion, accelerating mixing or even inhibiting it [80]. Whilst turbulence models and chemical mechanisms deal with turbulence and combustion individually, a turbulent combustion model is an additional requirement when simulating these conditions. Turbulent combustion models interweave the non-linear coupling effects of the chemistry and turbulence into the existing formulations used by CFD solvers [18].

3.4.1.2 Supercritical/Cryogenic Injection

Alongside the difficulties associated with modelling turbulent combustion, rocket engines yield extreme pressures and temperatures which force the contained flow into the supercritical regime, unlocking additional predictive complexities. When a fluid becomes supercritical, it shares the qualities of both a gas and a liquid, with the transition causing rapid non-linear changes in its thermo-physical properties [44]. The critical point is marked by a critical temperature and pressure (or density, see Tab.3.2), from which the widom line extends, marking the boundary at which the supercritical fluid becomes more "gas-like" or more "fluid-like". When the chamber pressure exceeds the critical pressure, propellants that are injected cryogenically (below the critical temperature), such as liquid oxygen, transition into a supercritcal state, moving across the widom line due to the sudden rise in temperature. This is called pseudo-boiling, characterised by the maxima in specific heat and drop in density, seen in Fig.3.8. Cutrone *et. at.* suggests that the injected propellant appears more like a turbulent gaseous jet than a liquid spray, as its inter-molecular forces diminish and it undergoes vaporisation, forming a continuous single phase mixture in the chamber [18].



(A) Oxygen specific heat transcriticality

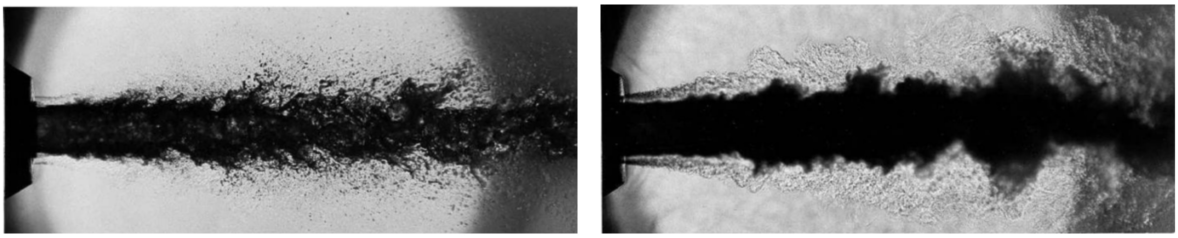
(B) Oxygen density transcriticality

FIGURE 3.8. Non-linearity surrounding the critical point of oxygen (66.6 bar)

TABLE 3.2. LOX/CH4 propellant critical points [76]

Propellant	**T$_c$ (K)**	**P$_c$ (bar)**	**D$_c$ (kg/m^3)**
Oxygen	154.6	50.4	436.1
Methane	190.6	46.0	162.7

The fluid structure of the supercritical injection can be visualised in Fig.3.9, in which Mayer *et al.* at DLR established a non-reacting setup with a single coaxial injection element [71]. A comparison was made between the cryogenic injection of liquid nitrogen into gaseuos helium at subcritical and supercritical chamber pressures. Fig.3.9a reveals the atomization process as a distinct droplet spray forms when the pressure is 1 MPa (10 bar). The aforementioned turbulent gaseous jet is clearly visible in Fig.3.9b, as the critical pressure is exceeded at



(A) Sub-critical chamber pressure

(B) Supercritical chamber pressure

FIGURE 3.9. Flow visualisation of a LN2/GHe single coaxial injection at sub and supercritical chamber pressures, taken from [71]

6 MPa (60 bar). Mayer *et al.* also saw the same trend with the LOX/H_2 combination [70]. Furthermore, research conducted by Chehroudi *et al.* explored the divergence angle of cryogencially injected fluids into a supercritical environment and how it compares to differing mixing layer flows. The results concluded a quantitative agreement with the theory of incompressible but variable density gaseous mixing layers, with subcritical cases aligning with liquid sprays [12].

Atomization followed by evaporation is a key step in turbulent spray combustion, however alongside Chehroudi *et al.*, both Oschwald *et al.* and Habiballah *et al.* found that the liquid propellant does not atomize in the supercritical realm, instead undergoing a diffusivity driven mixing process due to the negligible surface tension, highly sensitive to fluctuations in pressure and temperature [78, 12, 73, 41]. Whilst still being far from properly understood, the evolutionary process of liquid propellants that undergo transcritical injection is distinctly different from the classical dual-phase atomization and evaporation [19].

Numerically, the physical conditions of cryogenic and trans-supercritical injection require a single phase model as the thermo-physical properties remain continuous, seen in Fig.3.8. However, the sensitivity to temperature and pressure, with harsh non-linear gradients divorce the typical relationships given in the ideal gas law. Methods of accounting for this added complexity are explored in the following chapter, alongside formulations for turbulence, combustion and their coupled effects.

3.4.1.3 LOX/CH4 Combustion

The previous experimentation largely consists of non-reacting flows, with the exploration of supercritical LOX/CH_4 combustion gaining traction approximately 15 years ago [44]. As this propellant combination has been more seriously considered for high performance engines, such as the SpaceX Raptor and ESA Prometheus, rapid research and development emulating similar conditions has unfolded [67]. LOX/H_2 has historically been a popular choice, offering the highest performance. Thus, it serves as a common benchmark to compare with. At DLR in 2007, Yang *et al.*. utilised a single shear coaxial injector setup, using micro combustion chambers with optical access to explore the influence of methane's kinetics and properties [105]. Whilst only subcritical pressures were achievable, it was found that it was not sufficient to scale injector performance from hydrogen to methane. Lifted flames were present in most cases, yielding higher expansion angles, motivating plans to begin testing LOX/CH_4 in more detail, especially at supercritical pressures. A couple years later Lux and Haidn conducted LOX/CH_4 tests at 40-60 bar, covering both sub and supercritical pressure ranges whilst also researching the effects of recess in the injector, a design seen in BKN [59]. The steady state analysis of combustion roughness, with "rough" being defined as a pressure fluctuation of more than 5%, noted that recessing the LOX core made an improvement [95]. Time averaged images also noted higher flame expansion angles with a recessed core at supercritical pressures, with a strong correlation to the velocity and momentum flux ratios (Eq.3.2, Eq.3.3). Images captured through a small optical access window at the injection region used OH* radiation filters to capture the emission intensity, a technique also used with BKN. Again with recess, the intensity was greater due to improved mixing, with the flame anchored at the LOX post. This is backed by Théron *et al.* noting brighter OH* images as well as a shorter flame length and LOX jet [97]. An additional metric used was combustion efficiency, with the recess improving the result by 1%. The efficiency also rose with an increase in ROF (Eq.3.4), a relation also found by Soller *et al.* [92].

$$\mathrm{VR} = \frac{u_f}{u_o} \tag{3.2}$$

$$J = \frac{(\rho u^2)_f}{(\rho u^2)_o} \tag{3.3}$$

$$\text{ROF} = \frac{\dot{m}_o}{\dot{m}_f} \tag{3.4}$$

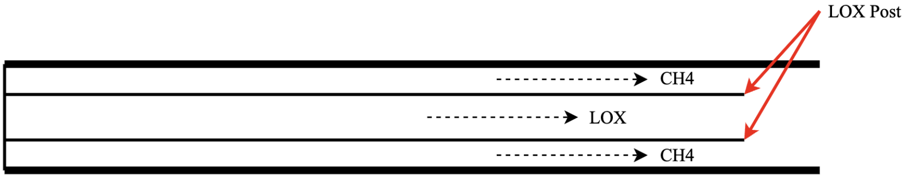


FIGURE 3.10. Cross section of a coaxial injector with a recessed LOX core

It is important to consider the element of "double transcriticality" in the Théron case, where both oxygen and methane are injected cryogenically and undergo a transition through the widom line. With similar ROF's Singla *et al.* compared this with gaseous methane injection, with only the liquid oxygen injected transcritically, an experiment used for the validation of many numerical studies discussed later. Again, chemiluminescence imaging of OH* as well as CH* radiation was utilised to characterise the flame topology, with significant variations in flame length and expansion found [90]. With gaseous methane the flame length appeared shorter, with a reduced LOX core. Additionally the shape remained cylindrical before expanding rapidly as the oxygen core was burnt, resembling LOX/H_2 cases. These effects are exacerbated due to the small ROF, with the increased methane flow rate with respect to oxygen enhancing the mass transfer from the oxygen core. Both Théron and Singla *et al.* employed sub 0.5 ROF's, far from the stoichiometric point of 4.

Employing an ROF closer to the stoichiometric point is more realistic, when considering the priority on fuel efficiency in high performance main stage engines [95]. However most engines run slightly fuel rich, accounting for imperfect combustion and the increased production of lighter exhaust products such as H_2, which in turn increases performance [16]. Arnold *et al.* tested a ROF of 3.4 across pressure ranges of 40-70 bar, whilst implementing fuel film cooling through tangential slots around the wall [7]. Effective wall temperature reduction was observed, with correlations arising between the circumferential positioning of these slots with respect to the injectors and the cooling efficiency. Additionally, increasing chamber pressure reduced the cooling efficiency, yet enhanced the mixing with the hot gases. A detailed wall heat flux test was conducted by Woodward *et al.* in which an ROF of 3.0 was used, with 3 varying injector types and momentum flux ratios [58]. Higher heat flux measurements were recorded with higher momentum flux ratios, attributed to improved mixing and combustion efficiency. Furthermore, an increasing chamber pressure yielded higher heat fluxes.

The most recent BKN test case will provide the experimental basis for this work [67]. Two load points were tested, both with a shear coaxial injector with a recessed LOX core. The ROF, velocity ratio, momentum flux ratios and chamber pressures are captured in Tab.6.4, with the remaining experimental details discussed throughout this thesis.

TABLE 3.3. BKN load point parameters

Parameter	**Load Point 1**	**Load Point 2**
ROF	2.8	3.0
VR	17.1	15.6
J	15.8	13.6
P (bar)	66.8	66.2

3.4.1.4 Transient Phenomena

The effects of transient phenomena that occur within combustion chambers is a major focus of DLR's Research Group for Combustion Dynamics. Ultimately, these effects can result in the catastrophic failure of an engine, starting as early as as the feed system [3]. Combustion instability remains one of the most challenging issues in rocket engine design, famously requiring the F-1 engine to go through over 2000 full engine tests resulting in numerous design changes to inhibit it[28]. Most notably, high frequency instabilities can arise due to coupling effects between fluid flow, heat release and chamber acoustic modes causing destructive pressure oscillations greater than 20% of the average chamber pressure [43]. The coupling mechanisms that give rise to these instabilities are not fully understood, yet they

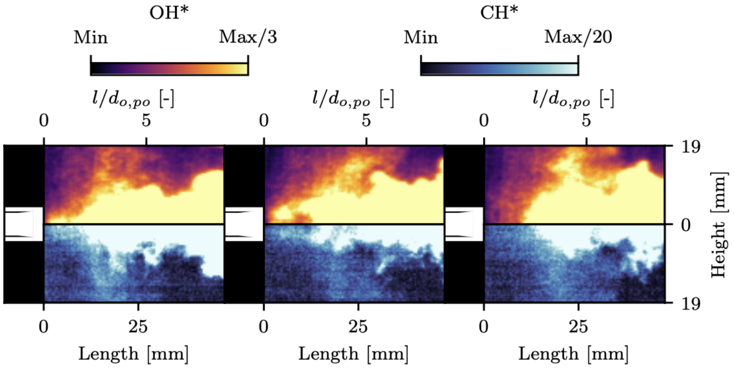


FIGURE 3.11. 3 different flame anchor positions captured through BKN's optical access window, taken from [67]

are an active avenue of pursuit at DLR Lampoldshausen in particular. Multi-injector setups, exploring both LOX/H_2 and LOX/CH_4 propellant combinations have found commonalities with LOX injector influence on high frequency instability. When using LOX/CH_4 a second case has been found, theorised to be driven by unstable flame anchoring [66, 6, 43]. Alongside a slower flame speed, the LOX/CH_4 chemistry is slower in comparison to LOX/H_2, influencing this hypothesis [32]. Hardi *et al.* noted unsteady pressure and temperature measurements indicative of this occurrence, however optical access would be required for confirmation [66]. BKN offers this ability, with Martin *et al.* noting unstable flame anchoring through a series of instantaneous flame images showing differing positions, see Fig.3.11 [67].

Whilst BKN's use for combustion instability analysis is important to note, numerically resolving transient phenomena is computationally expensive. In order reduce the computational cost, this work will use a time averaged approach to model the steady state mean flow field, disallowing the capture of any unsteady flow features. Hence, the effects of combustion instabilities will not be considered.

3.4.2 Numerical Process

With an understanding of the underlying physical processes and relationships discovered experimentally, the steps taken to numerically model a liquid rocket engine can be defined. From a high level, the entire process can be divided into 3 distinct steps [100]:

(1) **Pre-processing** - Model setup, definition of the simulation domain and grid, look-up table generation.

(2) **Numerical solution** - Scheme selection, equation discretisation, initialisation, boundary condition specification and computation.

(3) **Post-Processing** - Visualisation of the flow field, data processing and plotting.

Many CFD codes exist that facilitate this process with the ANSYS suite of software selected alongside MATLAB as a supplementary tool. Namely, ANSYS Fluent will be used as the code to calculate the numerical solutions. MATLAB will be used to aid in both pre and post processing due to its powerful data analysis capability. Further exploration into different software has not been undertaken, however DLR's in house solver TAU will be referenced as a source of result comparison.

The first choice to make is the numerical approach to turbulence. They differ greatly in their computational expense, dictated by their treatment of the governing equations.

Direct Numerical Simulations (DNS) - At the expense of enormous computational cost, DNS provides the most accurate approach for numerical calculations. With a direct approach to solving the Navier-Stokes equations without turbulence modelling, the solution is only limited by the accuracy of the numerical methods employed [56]. The lack of statistical turbulence modelling causes DNS to become too computationally demanding in most flow cases.

Large Eddy Simulations (LES) - A step down in computation expense from DNS is LES, an approach that directly computes large scale turbulence and models the smaller scale effects. With turbulent combustion, the large scale flame front can be computed instantaneously however sub-grid modelling is required for small turbulent scales and their effect on combustion. Unsteadiness and transience can be captured however the code resolves the flow in time, inflicting a computational cost 100-1000 times greater than RANS [80].

Reynolds Averaged Navier Stokes (RANS) - RANS is an effective method in capturing the statistical mean of the flow field and thus can be defined as time-averaged [56]. It can be likened to taking a long exposure image via a slow shutter speed as seen in Fig.3.12 [35]. As a result it is computationally efficient, yet it's limited by its inability to capture highly transient phenomena. The other relationships and important factors explored previously, namely heat transfer, flame shape and combustion efficiency can all be captured using RANS, making it a computationally viable method when transience isn't the core focus. RANS also allows for geometry simplification, including 2D asymmetric grids. The final computation yields a single converged solution disregarding time, providing mean results for all parameters [80]. RANS will be utilised in this work to model BKN.

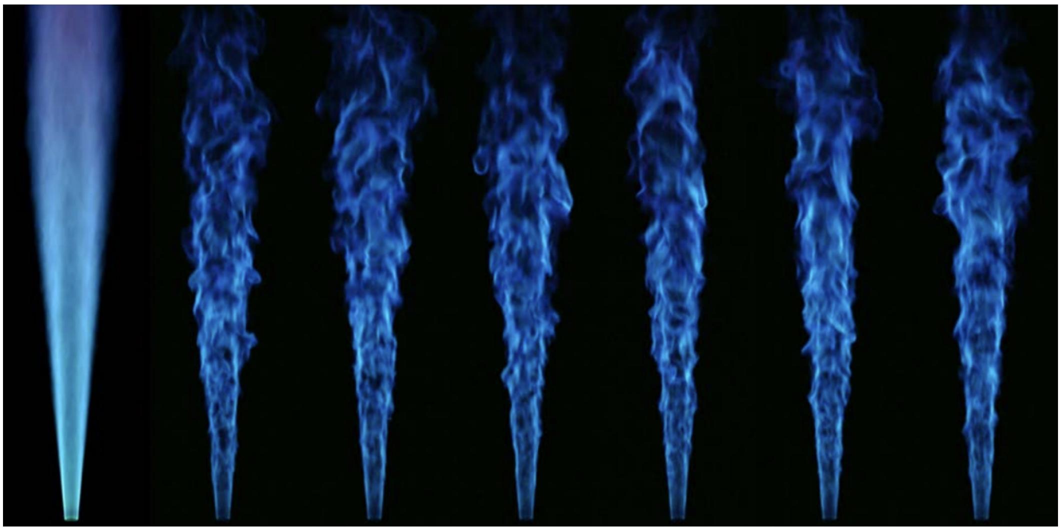


FIGURE 3.12. Chemiluminescence images of a turbulent jet flame, with the left a time-averaged visualisation through a slower shutter speed [35]

3.4.3 RANS Numerical Setup

The RANS numerical setup requires an entirely modelled approach as no direct instantaneous calculations are made. The following breakdown can be made, capturing the high level requirements of an appropriate setup. Note, the detailed mathematics of the following models are explored in the following case study chapters.

Turbulence Model - The turbulence model handles the flow dynamics of the system [80]. Numerous types exist, with differing mathematical formulations for the predictions of turbulent effects.

Chemical Mechanism - The chemical mechanism captures all the constituent species and their sequential elementary reactions encapsulated within the overarching combustion process.

Turbulent Combustion Model - The turbulent combustion model is required to embed the effects of the combustion reaction defined by the chemical mechanism into the turbulent flow field calculations.

Equations of State and Thermophysical Properties - The equation of state (EoS) is a thermodynamic description of the state of matter given the physical conditions, namely temperature and pressure [88]. As previously discussed, the thermophysical properties of fluids near the critical point undergo rapid change, captured more effectively by "real gas" equations of state.

Boundary Conditions - The simulation boundary conditions define the perimeters of the domain that influence the internal flow and ultimately differentiate one simulation from another. These include injection parameters, outlet parameters and wall treatment. Boundary conditions are matched with experimental setups for numerical validation.

3.4.4 Numerical Simulation of LOX/CH4 Combustion

The available literature pertaining to the RANS simulation of LOX/CH_4 combustion in single injector supercritical combustion chambers is scarce. In order to evaluate the current use of the different modelling techniques that would be implemented in this work, a broader range of operating conditions was considered for a comprehensive analysis.

In 2008 Cutrone *et al.* evaluated numerous models for LOX/CH_4 supercritical combustion, namely a comparison between ideal gas and real gas effects as well as wall function approaches [18]. Little difference was seen between wall function approaches whilst acknowledging their drawback with finer meshes. The most conclusive result was the ineffectiveness of ideal gas equations of state. Overestimations of up to 14% in chamber temperature and 17% in chamber pressure were noted with respect to the Peng Robinsion (PR) EoS, a popular real gas implementation. This is backed by Hickey *et al.* who also utilised the PR EoS for increased accuracy around the critical point in an LES simulation [44]. Alternatively, the development of the Redlich-Kwong (RK) EoS by Soave presents a more popular option for modelling high pressure combustors [91]. Now called the Soave-Redlich-Kwong (SRK) EoS, multiple

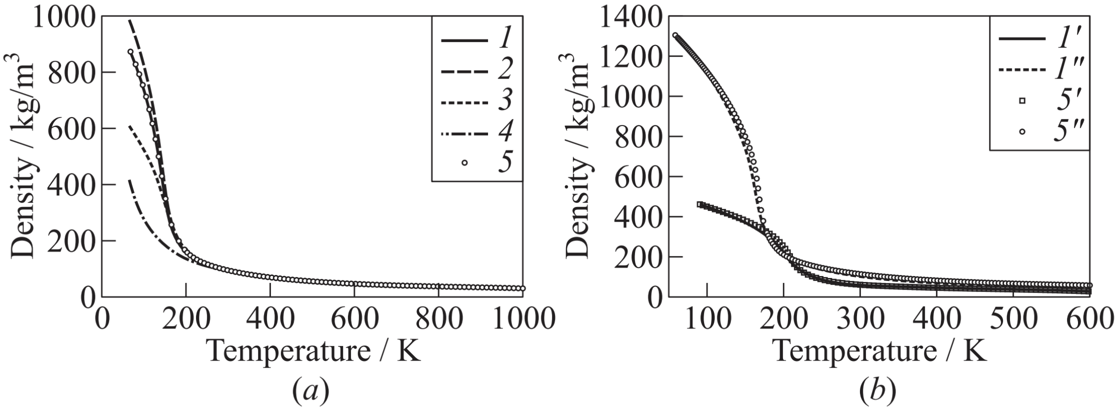


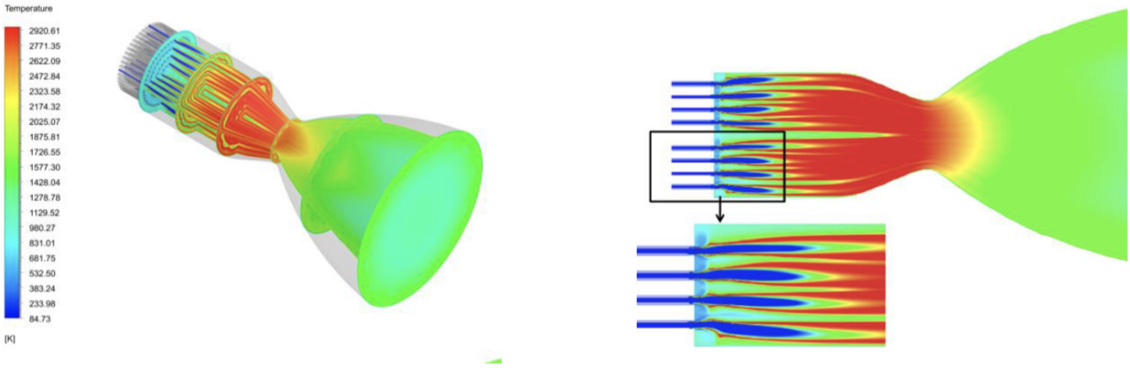
FIGURE 3.13. Evaluation of cubic EoS (1-SRK; 2-PR; 3-VDW; 4-PG) for density calculation and comparison with NIST data (5), p = 8.0 MPa: (a) N2 species and (b) CH4 (1'-SRK and 5'-NIST) and O2 species (1"-SRK and 5"-NIST), plots taken from [82]

works have implemented this method, yielding more accurate results. BKD, a multi-injector combustion chamber housed at DLR was simulated by both Chemnitz and Shulze *et al.*. Both utilised the SRK EoS, affirming its accurate conformation to NIST data, in particular for density [84, 14]. A more comprehensive study by Ribert *et al.* comparing 4 different EoS including PR and SRK confirms this. Density profiles of oxygen and methane over a large temperature range at supercritical pressures were analysed, seen in Fig.3.13 [82]. The SRK EoS maintained accuracy over the entire temperature range, including low temperatures down to 77.5 K, where other EoS broke down. The accuracy at low temperatures is supported by Oefelein [77].

Théron *et al.* conducted a numerical investigation to support the experimental test case of LOX/CH_4 cryogenic combustion in which the SRK EoS was adopted. An exploration into different chemical mechanisms was conducted with the complex GRI3.0 set as the benchmark for other comparisons [97]. Its acknowledged as a highly detailed scheme with 53 species and 325 reactions, however it lacks validation at high pressures [1]. DLR's reduced mechanism with 23 species and 51 reactions is also compared in this work, showing good conformity to axial temperature profiles and OH mass fractions. A reduced version of the GRI3.0, discounting the nitrogen compounds needed for air/methane combustion, was implemented by Priyadarshini *et al.* and Di Giorgi *et al.* [81, 21]. Both numerical studies compared their results to the previously discussed supercritical LOX/CH_4 experimental test case by Singla *et al.* referred to as "G2" [90]. An even simpler mechanism called the Jones-Lindstedt (JL) scheme, consisting of just 9 species and 6 reactions was used as a source of comparison [36]. Both found a minor accuracy gain with the reduced GRI3.0, however Di Giorgi noted that in conjunction with the PDF-flamelet turbulent combustion model, the reduced GRI3.0 overestimated flame length and peak axial temperature positioning slightly, whereas the JL mechanism underestimated both by the same amount. When using the eddy dissipation concept (EDC) approach for turbulent combustion, both produced near identical and accurate results.

Through the investigation of different turbulent combustion methods it became clear that the approach had a far bigger impact than the chemical mechanism itself. Previous papers by Sciolti and Di Girogi *et al.* explore different numerical options, maintaining the conclusion that the EDC model provides the most accurate results regarding flame length and shape [22, 85]. Whilst more detailed mechanisms performed slightly better, the computational cost was not justified. Likewise, the EDC method, whilst producing noticeable accuracy, induces a computational cost far higher than methods such as the equilibrium PDF and PDF-flamelet approaches. Between the two, negligible differences were observed yet an acceptable conformity to experimental results was concluded. Supporting this tradeoff with computational expense is the work by Sharma *et al.* in which a supercritical multi-injector LOX/CH_4 study was conducted. Flow field results are seen in Fig.3.14 [88]. It is important to note, these two approaches do not account for real gas affects, with numerical corrections made later in Fluent [51]. Wu *et al.* addresses the lack of thermal boundary conditions within the PDF-flamelet formulation, noting significant wall heat flux improvements when compared to its original formulation [61].

When considering the supercritical injection of liquid oxygen, its thermophysical properties undergo a rapid yet continuous change, seen in Fig.3.8. This implies that a single phase approach may be applicable, confirmed by the numerical work by Di Giorgi *et al.* in 2014 [21]. When considering subcritical pressures, a multi-phase model that distinguishes the



(A) 3D temperature contour

(B) 2D temperature contour

FIGURE 3.14. Temperature contours of the LOX/CH_4 numerical simulation conducted by Sharme et al. [88]

liquid and gas states of oxygen was required to yield accurate results however it was presented that either worked in the supercritical regime.

Looking back at the work by Priyadarshini *et al.*, four different turbulence models were tested. These include 2 versions of the k-ϵ (standard and realizable), the Reynolds stress model (RSM) and the k-ω SST model [81]. Again the G2 case was used as the source of experimental validation. Only the standard k-ϵ model was able to accurately reproduce the flame length and shape, accompanied by an altered model constant. As seen in Fig.3.15, the others all overestimated the flame length significantly. The precise axial point of peak temperature aids in the result, with the peaks of all other models further downstream.

Similar comparisons were made by Shaikh *et al.* and Mardani *et al.* however an inclusion of the RNG k-ϵ model was made, another modification of the standard k-ϵ model [64, 87]. Mardani saw accurate results utilising either the RNG or standard variations however Shaikh was able to affirm the superiority of the standard model in conjunction with the PDF-flamelet turbulent combustion model. It should be noted that in the Shaikh case, the oxygen was injected as a gas. Moving further from BKN conditions, a supercritical non-reacting nitrogen flow was modelled using 6 difference turbulent models by Magalhães *et al.* and noted the greatest accuracy was seen with the RNG model [62]. The standard k-ϵ still remains the most popular choice for reactive cases, chosen in most of the previously discussed works. An interesting new turbulence model developed by ANSYS, called the generalised k-ω model

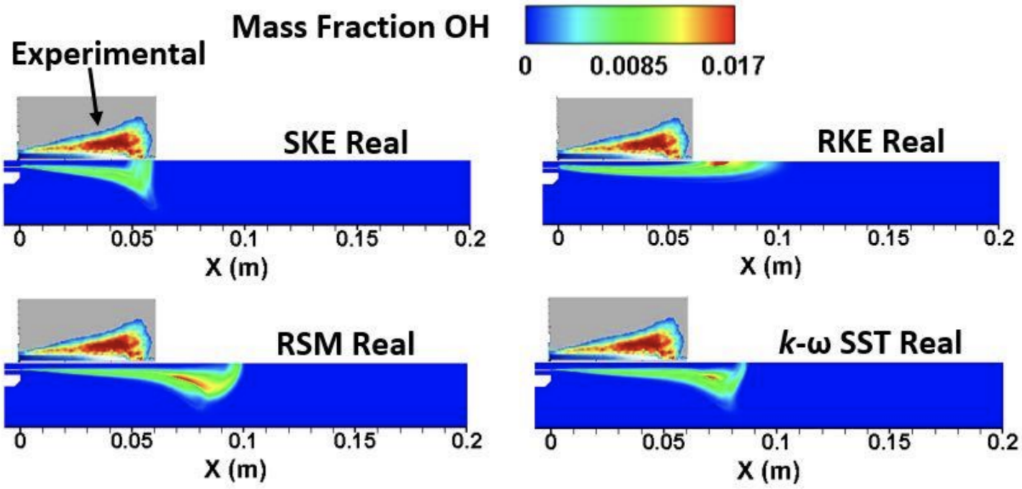


FIGURE 3.15. Comparison between different turbulent models and their conformity to experimental data, taken from [81]

(GEKO) presents a way to easily switch between models through a set of flexible parameters [39]. Whilst there is no resource that provides a mathematical basis for the parameters, Strokach *et al.* investigated their effect on a GOX/CH_4 single injector supercritical combustion chamber [94]. Affirming the accuracy of the standard k-ϵ model, parameters that emulated it saw the closest conformity to experimental data. The parameter CMIX, which optimizes free shear flow mixing, was increased further to better match wall heat flux and pressure experimental data, whilst maintaining other standard k-ϵ equivalent parameters [38].

When dissecting the mathematics of the turbulence models, the turbulent Schmidt number has a large effect on the flame structure. Chemnitz conducted a thorough study, investigating numbers in the range of 0.5 to 1 [84] with the standard k-ϵ model. Flame length was seen to decrease with a lowering of the Schmidt number, with harsher pressure gradients attributed to enhanced mixing and combustion. In accordance to pressure sensors, values of 0.5-0.6 were the most accurate. This is affirmed with comparison to OH* imaging, with the flame structure aligning best at a value of 0.5. However it was acknowledged that 0.5 is already considered fairly low and thus should be treated with caution. This is supported by Keller *et al.*, whose work saw wall pressures at a value of 0.7 show a steeper but more accurate gradient in comparison to 0.9 and 1.1 [15]. Furthermore, Ivancic *et al.* observed a smaller outer circulation zone with a decreasing turbulent Schmidt number, seen in Fig.3.16, again

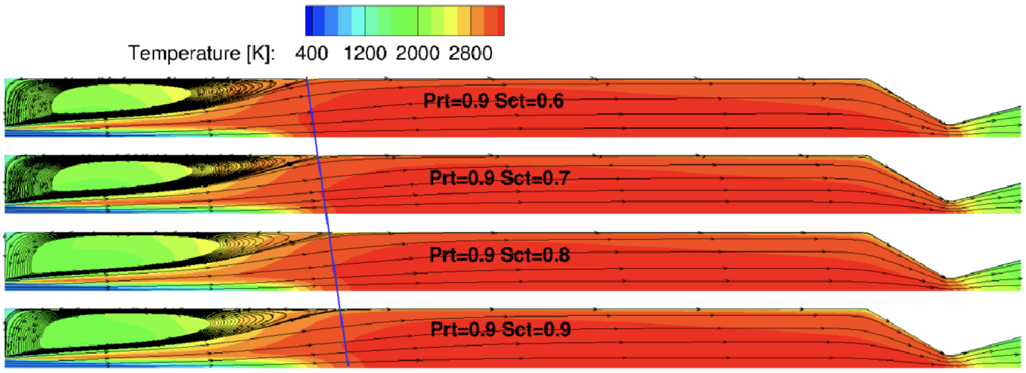


FIGURE 3.16. Temperature contours with a variable turbulent Schmidt number, taken from [52]

attributed to intensified combustion processes [52]. Finally, a study by Hwang *et al.* pursued a variation of turbulent Schmidt numbers in a LOX/H_2 combustor, with mid-range values of 0.4-0.6 presenting the highest degree of accuracy [46].

Looking back at the wall function approach used by Cutrone *et al.* it is noted that empirical formulas are used to bridge the boundary flow at the wall, which are sensitive mesh resolution [18]. The work by Chemnitz on BKD was done in ANSYS Fluent and employed 2 methods that are not mesh sensitive, referred to as enhanced wall treatment (EWT) and the Menter Lechner method. The wall pressure and heat release results were nearly identical, with Menter Lechner yielding a small reduction in flame length [84]. Hong *et al.* noted the importance of using EWT for boundary flows with regard to the correct heat flux. Fiuza and Rezende utilised standard wall functions, EWT and Menter Lechner wall treatments with the standard *et al.* model [33] with EWT displaying the highest fidelity.

Zhukov noted the numerical boundary conditions and mixture transport quantities had a significant impact on the wall heat flux measurements [108]. Mixed approaches were seen when setting wall boundary conditions, with Hong *et al.* using a fixed temperature profile, however adiabatic and isothermal conditions were present in previous works for simplicity [106]. Limited literature was found that compared these setups. As for injection initial conditions, French *et al.* noted higher heat fluxes with lower ROF's, consistent with experimental works discussed previously [37].

Tab.3.4 encapsulates the discussed numerical work, in particular noting the adopted equation of state (EoS), chemical mechanism (CM), turbulence model (TM) and turbulent combustion model (TCM). N/A is used when the method is not applicable or not stated.

TABLE 3.4. Summary of the models employed by the discussed numerical studies

Author	**Solver**	**Type**	**EOS**	**CM**	**TM**	**TCM**	**Propellants**	**Reference Case**
Cutrone et al.	C3NS	RANS	PR	RJL	SKE	PaSR/EDC	LOX/CH4	RCM01-b, LES
Hickey et al.	In-house	LES	PR	Burke	N/A	FPV	GO2/GH2	Pen-state
Chemnitz	Fluent	RANS	SRK	N/A	SKE	PDF-FLA	LOX/CH4	BKD
Shulz et al.	N/A	RANS	SRK	N/A	N/A	N/A	LOX/H2	BKD
Ribert et al.	In-house	N/A	PR, SRK, VDW, PG	N/A	N/A	N/A	LOX/CH4	NIST
Théron et al.	In-house	RANS	SRK	GRI3.0, DLR	SKE	Arrhenius	LOX/CH4	BHP
Priyadarshini et al.	Fluent	RANS	SRK	RGRI3.0, RJL	SKE, RKE, RSM, SST	PDF-FLA	LOX/CH4	G2
Di Giorgi et al.	Fluent	RANS	SRK	RGRI3.0, RJL	RKE	EDC, PDF-FLA, PDF-EQ	LOX/CH4	G2
Sciolti	Fluent	RANS	SRK, IG	RGRI3.0, RJL	RKE	EDC, PDF-FLA, PDF-EQ	LOX/CH4	G2
Sharma et al.	In-house	RANS	SRK	ARM-19	SST	EDC, PDF-FLA	LOX/CH4	G2
Shaikh et al.	Fluent	RANS	N/A	N/A	SKE, RKE, RSM, RNG	PDF-FLA, PDF-EQ	GOX/CH4	N/A
Mardani et al.	In-house	RANS	IG	Burke	SKE, RNG	EDC	LOX/H2	Mascotte
Strokach et al.	Fluent	RANS	N/A	DLR, RAMEC	GEKO	PDF-FLA	GOX/CH4	TUM
Keller et al.	Fluent	RANS	IG	DLR, GRI3.0	SKE, SST	PDF-FLA	GOX/CH4	SFB-TR40
Ivancic et al.	CFX, TAU, Rocflam 3	RANS	SRK	Gaffey	SST, RSM, SKE	PDF-FLA	GO2/H2	Pen-state
Hwang et al.	In-house	LES	RK-PR	N/A	N/A	SLFM	LOX/H2	RCM-3
Zhukov et al.	CFX	RANS	IG	N/A	SST	EDM	GOX/H2	Pen-state
Hong et al.	Fluent	RANS	IG	RGRI3.0	SKE	EDC	GOX/CH4	TUM
French et al.	Fluent	RANS	IG	RJL	SST	EDC	LOX/CH4	Pen-state

3.5 Key Findings

Through the investigation into the development of liquid rocket engines, their design and the experimental/numerical work to date, a wealth of useful information has been gleaned. Summarised as follows, the next steps in developing the numerical simulation of BKN have been well informed.

- Non-reacting flows in the supercritical regime present mixing structures that resemble turbulent gaseous jets without distinct droplet atomization and evaporation processes.
- Thermo-physical properties of a fluid undergo rapid non-linear changes when undergoing a trans-supercritical change of state, passing through the widom line.
- In reactive cases, strong coupling between the turbulent flow field and combustion flame exists within the thrust chamber.
- Injection parameters such as the ROF, velocity ratio and momentum flux ratio have noticeable impacts on flame length, structure and wall heat flux measurements.
- Characterisation of the flame topology is attempted through chemiluminscence imaging of OH and CH radiation, seen through small optical access windows. BKN poses an advantage with a window that spans near the entire length of the chamber.
- The choice of turbulence model has a large impact on the flame length and shape, with the standard k-ϵ model being the most popular and accurate for "BKN-like" conditions. The RNG k-ϵ model also presents as a potential comparison, however it is not abundant in existing literature.
- Within the turbulence model of choice, constants have been altered for a better match to experimental data. The most notable was the turbulent Schmidt number. Mid ranges numbers of 0.4-0.6 are already considered fairly low however they consistently produced accurate results in comparison to experimental data. A priority should be placed on other facets of the simulation before tuning this parameter.
- Through both experimental and numerical analysis it is clear ideal gas laws are not applicable to supercritical combustion. The SRK EoS has shown the best conformity to NIST data for oxygen and methane over large temperatures ranges whilst also yielding the most accurate results in numerical studies.

- The chosen chemical mechanism has seen to have a relatively minimal effect on the accuracy, with more detailed schemes improving fidelity marginally. Most studies used reduced schemes for the sake of computational expense, with the reduced GRI3.0 forming the common benchmark. DLR's reduced scheme and the simple JL scheme are other notable mentions.
- The method of modelling turbulent combustion in literature consisted nearly entirely of either the EDC or PDF-flamelet approaches. Often the two were compared, with EDC consistently proving to be the most accurate. The computational savings using the PDF-flamelet method are acknowledged, with the degradation in accuracy deemed a worthwhile trade-off in many cases.
- The PDF-flamelet method utilises many flow assumptions, with underestimated wall heat flux noted as a consequence.
- Limited literature exists that analyses the effects of different wall boundary conditions and near-wall treatment. One key finding was that standard wall functions produce errors in the boundary layer.

The findings described above provide a reference point into which a more detailed exploration into model formulations can be made with respect to their applicability to BKN. Whilst some methods can be discounted purely based the review of literature, it is clear there is no perfect combination thus further comparisons need to be made. As the aim is to formulate a numerical setup with the highest degree of fidelity possible, a sequential analysis of all the components that construct the overall simulation is required.

CHAPTER 4

Case Study 1 - Computational Fluid Dynamics

This chapter presents the governing equations and models for the CFD configuration utilised in this work. As a case study, it captures the course content of AMME5202 with relevance to simulation of BKN. Section 4.1 presents the governing equations for general fluid flow, with Section 4.2 introducing the modifications for a reactive flow. Section 4.3 derives the Reynolds Averaged Navier Stokes equations. Section 4.4 breaks down the methods of modelling turbulence. Section 4.5 introduces equations of state and formulations that suit supercritical conditions. Finally, Section 4.6 discusses the computational setup and domain facilitating the numerical calculations.

4.1 Governing Equations

The governing equations describing fluid flow are referred to as the Navier Stokes equations. In the same way Newton's second law allows the calculation of velocity of a solid body per the acting forces, the Navier Stokes equations describe the same for a finite volume of fluid. More formally, the momentum of the fluid is equal to the sum of the external forces acting upon the fluid. This can be expressed in concise vector form as follows:

$$\frac{D(\rho \boldsymbol{U})}{Dt} = \boldsymbol{f} \tag{4.1}$$

Here, ρ is the density of the fluid parcel. As momentum changes with both time and space, the total derivative term $\frac{D}{Dt}$ can be expanded into the temporal and spacial derivatives as follows:

$$\frac{D}{Dt} = \underbrace{\frac{\partial}{\partial t}}_{\text{Time}} + \underbrace{U_x \frac{\partial}{\partial x} + U_y \frac{\partial}{\partial y} + U_z \frac{\partial}{\partial z}}_{\text{Space}} \tag{4.2}$$

The velocity vector $\boldsymbol{U}$ can be expressed in einstein notation, the convention used henceforth. A single index is used to capture the 3 velocity directions, in which a repeated index is summed over as such:

$$\boldsymbol{U} = u_i \text{ where } i = 1, 2, 3 \; (x, y, z) \tag{4.3}$$

$$u_i u_i = \sum u_i u_i = \sum_{i=1}^{3} u_i u_i \tag{4.4}$$

When applying these conventions to Eq.4.1, the momentum equation is formally defined as:

$$\frac{\partial}{\partial t}\rho u_j + \frac{\partial}{\partial x_j}\rho u_i u_j = \frac{\partial \sigma_{ij}}{\partial x_i} + \rho g \tag{4.5}$$

$$\sigma_{ij} = \underbrace{-p\delta_{ij}}_{\text{Pressure tensor}} \underbrace{-\frac{2}{3}\mu \frac{\partial u_k}{\partial x_k}\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)}_{\text{Viscous tensor } (\tau_{ij})} \tag{4.6}$$

$$\delta_{ij} = 1 \text{ if } i = j, \; 0 \text{ otherwise} \tag{4.7}$$

The sum of the external forces has been expanded on the right hand side. The Cauchy stress tensor σ_{ij} is a combination of the pressure and viscous tensors which accounts for the pressure and shear/normal forces respectively. The third term g accounts for external volumetric forces exerted on the fluid parcel. Here μ is the dynamic viscosity, p is the pressure and δ is known as the Kronecker symbol, defined in Eq.4.7.

In addition the momentum equations, mass is also conserved, yielding the mass continuity equation below:

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \tag{4.8}$$

Here, the first term encompassed the change in mass with respect to time, with the second term describing the mass flow in and out of the fluid element.

The momentum in each spatial direction, alongside the mass continuity equation define 4 of the 5 classical Navier Stokes equations. The final equation, referred to as the conservation of energy, can take multiple forms. In the context of this work, it exudes a dependence on reactive flow modifications, hence its introduction in the following section.

4.2 Reactive Flows

When considering the Navier Stokes equations in a reactive flow, additional terms need to be added. A key primitive variable is the individual species mass fraction, defined in Eq.4.9, where subscript k is used to denote species 1 through N and m is the mass in the given volume.

$$Y_k = \frac{m_k}{m} \tag{4.9}$$

The volumetric parcel of fluid can be viewed as the sum of the individual species composing it, thus the external force term g in Eq.4.5 can be re-written as such:

$$\frac{\partial}{\partial t}\rho u_j + \frac{\partial}{\partial x_j}\rho u_i u_j = \frac{\partial \sigma_{ij}}{\partial x_i} + \rho \sum_{k=1}^{N} Y_k f_{k,j} \tag{4.10}$$

Additionally, the mass continuity equation can be written for each individual species in terms of the mass fraction. This increases the set of Navier Stokes equations by N species, and thus the computational demand. The primary method of modelling combustion in this work, omits these equations through the introduction of the mixture fraction, discussed in Chapter 5.

$$\frac{\partial \rho Y_k}{\partial t} + \frac{\partial}{\partial x_i}(\rho(u_i + V_{k,i})Y_k) = \dot{\omega}_k \quad \text{for } k = 1, N \tag{4.11}$$

Here, V_{ki} is the i'th component of the diffusion velocity V_k of species k and $\dot{\omega}$ is its reaction rate, accounting for the generation or loss of species during the combustion reaction. Refer to Chapter 5 for more detail on chemical kinetics.

Finally, the conservation of energy equation is presented in Eq.4.12. Multiple forms of this equation exist, however here it is expressed as the sum of the sensible and kinetic energy.

$$\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x_i}(\rho u_i E) = \dot{\omega}_T - \frac{\partial q_i}{\partial x_i} + \frac{\partial}{\partial x_j}(\sigma_{ij} u_i) + \dot{Q} + \rho \sum_{k=1}^{N} Y_k f_{k,i}(u_i + V_{k,i}) \tag{4.12}$$

$$E = e_s + \frac{1}{2} u_i u_i \quad \text{where} \quad e_s = \int_{T_0}^{T} C_v dT - RT_0/W \tag{4.13}$$

When breaking down the equation, the first term describes change in total energy E (sensible (e_s) and kinetic) with time. Sensible energy is expressed as the integral of mixture heat capacity C_v. The second term encapsulates the total energy change with respect to fluid flow through 3 dimensional space. The right hand side captures the heat release due to combustion $\dot{\omega}_T$, the energy flux q_i, the power produced by the pressure, stress and volume forces and an additional heat source term $\dot{Q}$. This term can be attributed to ignition heat or radiative flux.

Thus far the presented equations are the instantaneous Navier-Stokes. To remove the dependence on time for a mean flow field solution, the process of Reynolds decomposition is undertaken. Additionally, a series of closures, most notably the turbulence model and EoS are required to apply these equations to numerical calculations. All of these methods are discussed henceforth.

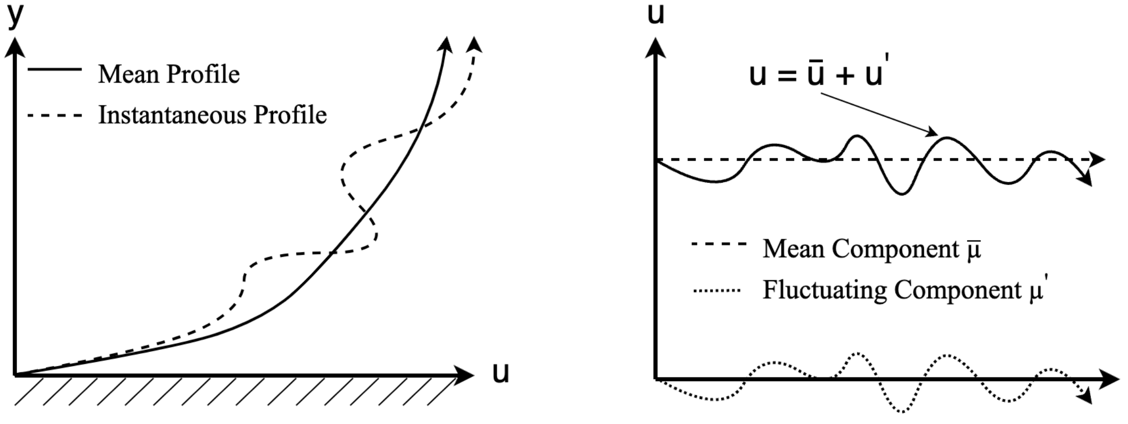
4.3 Reynolds Decomposition

For the sake of computational feasibility, this work utilises the RANS equations to capture a time averaged CFD solution. RANS equations are derived through the decomposition of the instantaneous Navier-Stokes into mean and superimposed fluctuating components. The velocity term is seen in Eq.4.14 with $\bar{u}_i$ and $u_i^{'}$ denoting the mean and fluctuating components respectively. A visual representation of these values is depicted in Fig.4.1. The mean component is defined in Eq.4.15 which can be used to derive the mean of the fluctuating component, proven to be 0 in Eq.4.16.

$$u_i = \bar{u}_i + u_i^{'} \tag{4.14}$$

$$\bar{u}_i \;=\; \frac{1}{\tau}\int_0^{\tau} u_i dt \;=\; \frac{1}{\tau}\int_0^{\tau} (\bar{u}_i + u_i^{'})dt \;=\; \frac{\bar{u}_i}{\tau}\int_0^{\tau} dt + \frac{1}{\tau}\int_0^{\tau} u_i^{'}dt \tag{4.15}$$

$$\bar{u}_i = \bar{u}_i + \bar{u_i^{'}} \quad \rightarrow \quad \bar{u_i^{'}} = 0 \tag{4.16}$$



(A) Instantaneous vs mean velocity profile

(B) Mean/fluctuating components of velocity

FIGURE 4.1. RANS decomposition into mean and fluctuating components for a time averaged profile

An additional rule to consider in the decomposition process is the relationship between terms. Density is averaged in the same manner (Eq.4.17), however it is usually seen in conjunction with the velocity term. When considering the multiple of the mean component of density and the fluctuating component of velocity, Eq.4.18-4.19 proves this to be 0.

$$\rho = \bar{\rho} + \rho^{'} \tag{4.17}$$

$$\bar{u}\rho^{'} = \bar{u}\rho - \bar{u}\bar{\rho} \tag{4.18}$$

$$\overline{\bar{u}\rho'} = \overline{\bar{u}\rho} - \overline{\bar{u}\bar{\rho}} = \bar{u}\bar{\rho} - \bar{u}\bar{\rho} = 0 \tag{4.19}$$

These rules become essential when the decomposition and time averaging is applied to the Navier Stokes equations. The momentum equation (Eq.4.10) contains the Cauchy stress tensor, which is expanded to produce Eq.4.20. Note, the external volume forces are neglected.

$$\frac{\partial}{\partial t}\rho u_j + \frac{\partial}{\partial x_j}\rho u_i u_j = \frac{\partial}{\partial x_i}\left[-p\delta_{ij} - \frac{2}{3}\mu\frac{\partial u_k}{\partial x_k}\delta_{ij} + \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \tag{4.20}$$

Time averaging and decomposing the left hand side yields:

$$\overline{\frac{\partial}{\partial t}(\bar{\rho} + \rho')(\bar{u_j} + u_j^{'})} + \overline{\frac{\partial}{\partial x_j}(\bar{\rho} + \rho')(\bar{u_j} + u_j^{'})(\bar{u_i} + u_i^{'})} \tag{4.21}$$

On expansion, terms that are equal to 0 are identified, marked with a slash in Eq.4.22.

$$\begin{gathered}
\frac{\overline{\partial\bar{\rho}\bar{u}_j}}{\partial t} + \cancel{\frac{\overline{\partial\rho'\bar{u}_j}}{\partial t}} + \cancel{\frac{\overline{\partial\bar{\rho}u_j^{'}}}{\partial t}} + \bcancel{\frac{\overline{\partial\rho' u_j^{'}}}{\partial t}} + \frac{\overline{\partial\bar{\rho}\bar{u}_j\bar{u}_i}}{\partial x_j} + \cancel{\frac{\overline{\partial\bar{\rho}\bar{u}_j u_i^{'}}}{\partial x_j}} + \\
\cancel{\frac{\overline{\partial\bar{\rho}u_j^{'}\bar{u}_i}}{\partial x_j}} + \frac{\overline{\partial\bar{\rho}u_j^{'}u_i^{'}}}{\partial x_j} + \cancel{\frac{\overline{\partial\rho'\bar{u}_j\bar{u}_i}}{\partial x_j}} + \bcancel{\frac{\overline{\partial\rho'\bar{u}_j u_i^{'}}}{\partial x_j}} + \bcancel{\frac{\overline{\partial\rho' u_j^{'}\bar{u}_i}}{\partial x_j}} + \bcancel{\frac{\overline{\partial\rho' u_j^{'}u_i^{'}}}{\partial x_j}}
\end{gathered} \tag{4.22}$$

Here, a differentiation is made between the forward and back-slashed terms. The forward slash marks terms equal to 0 as per Eq.4.19. The back-slashed terms contain the multiple of the fluctuating density with up to 2 fluctuating velocity terms, which in constant density flows can be considered to equal 0. With the same process applied to the right hand side, the RANS momentum equation can be expressed as:

$$\frac{\partial \bar{\rho}\bar{u}_j}{\partial t} + \frac{\partial \bar{\rho}\bar{u}_j\bar{u}_i}{\partial x_j} + \frac{\partial \bar{\rho}u_j^{'}u_i^{'}}{\partial x_j} = \frac{\partial \bar{p}}{\partial x_i} - \frac{\partial}{\partial x_j}\mu\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} - \frac{2}{3}\frac{\partial \bar{u}_k}{\partial x_k}\right) \tag{4.23}$$

When considering the supercritical injection of LOX, the density rapidly changes (see Fig.3.8b), rendering the assumption applied to back-slashed terms invalid. A technique called Favre averaging takes the mass weighted average of each quantity in order to account for unclosed correlations between the fluctuating density and other terms. Similarly to before, any quantity f can be split into its mean and fluctuating component, alongside the definitions given in Eq.4.25.

$$f = \tilde{f} + f^{''} \tag{4.24}$$

$$\tilde{f} = \frac{\overline{\rho f}}{\bar{\rho}} \quad \text{and} \quad \tilde{f''} = 0 \tag{4.25}$$

It is important to note that density and pressure remain Reynolds averaged. The Favre averaging process follows the same procedure as Reynolds averaging in which the decomposition rules are applied and the time average is taken. Using these definitions the mass, momentum, energy and species conservation equations can be derived, presented as follows:

Conservation of Mass

$$\frac{\partial \bar{\rho}}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_i}{\partial x_i} = 0 \tag{4.26}$$

Conservation of Momentum

$$\frac{\partial\bar{\rho}\tilde{u}_i}{\partial t}+\frac{\partial}{\partial x_i}(\bar{\rho}\tilde{u}_i\tilde{u}_j)+\frac{\partial\bar{p}}{\partial x_j}=\frac{\partial}{\partial x_i}\left(\tau_{ij}-\bar{\rho}\widetilde{u_i''u_j''}\right) \tag{4.27}$$

Chemical Species

$$\frac{\partial\bar{\rho}\tilde{Y}_k}{\partial t}+\frac{\partial}{\partial x_i}(\bar{\rho}\tilde{u}_i\tilde{Y}_k)=-\frac{\partial}{\partial x_i}\left(\overline{V_{k,i}Y_k}+\bar{\rho}\widetilde{u_i''Y_k''}\right)+\overline{\dot{\omega}_k}\quad\text{for}\quad k=1,N \tag{4.28}$$

Conservation of Energy

$$\frac{\bar{\rho}\tilde{h}_s}{\partial t}+\frac{\partial}{\partial x_i}(\bar{\rho}u_i\tilde{h}_s)=\overline{\dot{\omega}_T}+\frac{\overline{Dp}}{Dt}+\frac{\partial}{\partial x_i}\left(\overline{\lambda\frac{\partial T}{\partial x_i}}-\rho\widetilde{u_i''h_s''}\right)+\overline{\tau_{ij}\frac{\partial u_i}{\partial x_j}}-\frac{\partial}{\partial x_i}\left(\overline{\rho\sum_{k=1}^{N}V_{k,i}Y_kh_{s,k}}\right) \tag{4.29}$$

Here the conservation of energy is presented in terms of sensible enthalpy h_s, an alternate form to Eq.4.12. Fluent uses the enthalpy form of the energy equation when non-premixed combustion is modelled, expanded on in Chapter 5. A more detailed derivation and description of the terms and relationships in the presented Favre-averaged equations is given by Poinsot [80]. Within these equations, new unclosed terms are formed which require additional modelling, rendering the current system of equations unsolvable. In particular, the Reynolds stresses $(\widetilde{u_i''u_j''})$ necessitate the introduction of a turbulence model, discussed in the following section.

4.4 Turbulence Modelling

The introduction of a turbulence model has the purpose of relating the unclosed Reynolds stress term to the mean flow variables, closing the overall system of governing equations. Many types exist, including algebraic, 1 equation and 2 equation methods. A series of different turbulence models implemented in numerical studies similar to BKN were reviewed in Chapter 3. It was concluded that the standard k-ϵ model (2 equation) matched experimental results the closest, with the "renormilisation group" (RNG) variant as a potential alternative. Finally the GEKO model developed by ANSYS offers a flexible choice, albeit mathematically ambiguous, designed to emulate different models through a set "free" parameters. Hence, these models will be secondary sources of comparison, with the standard k-ϵ adopted as the primary choice.

4.4.1 Standard $k - \epsilon$ Model

The standard k-ϵ is a class of eddy-viscosity model. This type of model is underpinned by the Boussinesq hypothesis which states that the Reynolds stresses are proportional to the velocity gradient in a similar manner to the viscous stresses. This can be depicted visually in Fig.4.2 which illustrates the mean velocity profile for a wall shear flow. The faster velocities of the particles above accelerate the slower particles below them, forming a net transfer of momentum downwards, or in the direction of the velocity gradient.

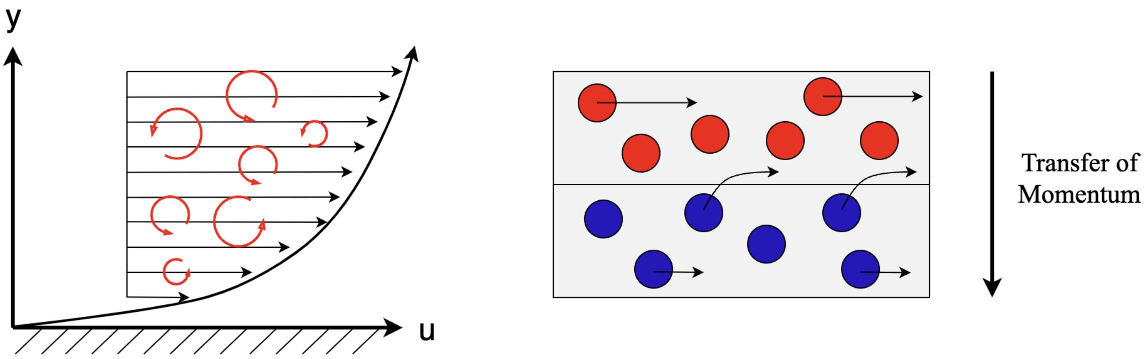


FIGURE 4.2. Boundary shear flow illustration of the Bousinesq hypothesis

Mathematically, the Reynolds stress term is related to the velocity gradient via the turbulent viscosity μ_t, a new term that requires its own definition and modelling. It controls the strength of diffusion induced by the transfer of momentum due to turbulence, with a higher value indicative of larger turbulent fluctuations. It is usually considered using the viscous tensor τ_{ij}, which has been expanded in Eq.4.30. The turbulent kinetic energy k is also introduced as an additional term.

$$\bar{\rho}\widetilde{u_i''u_j''} = -\mu_t \left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} - \frac{2}{3}\delta_{ij}\frac{\partial \tilde{u}_k}{\partial x_k} \right) + \frac{2}{3}\rho k \tag{4.30}$$

In order to close the Reynolds stress term, two transport equations are used, one which describes the turbulent kinetic energy k and one that describes its dissipation rate ϵ. The equation for k was derived mathematically whereas the ϵ equation was formulated using physical reasoning. Using the standard $k - \epsilon$ approach, the turbulent viscosity is estimated in Eq.4.31 with the transport equations for k and ϵ given in Eq.4.32 and Eq.4.33 respectively.

$$\mu_t = \bar{\rho} C_\mu \frac{k^2}{\epsilon} \tag{4.31}$$

$$\frac{\partial}{\partial t}(\bar{\rho} k) + \frac{\partial}{\partial x_i}(\bar{\rho}\tilde{u}_i k) = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x_i}\right] + P_k - \bar{\rho}\epsilon \tag{4.32}$$

$$\frac{\partial}{\partial t}(\bar{\rho}\epsilon) + \frac{\partial}{\partial x_i}(\bar{\rho}\tilde{u}_i \epsilon) = \frac{\partial}{\partial x_i}\left[\left(\mu + \frac{\mu_t}{\sigma_\epsilon}\right)\frac{\partial \epsilon}{\partial x_i}\right] + C_{\epsilon 1}\frac{\epsilon}{k}P_k - C_{\epsilon 2}\bar{\rho}\frac{\epsilon^2}{k} \tag{4.33}$$

$$P_k = \bar{\rho}\widetilde{u_i^{''} u_j^{''}}\frac{\partial \tilde{u}_i}{\partial x_j} \tag{4.34}$$

The source term P_k represents the production of turbulent kinetic energy, relating the Reynolds stress term to the velocity gradient, drawn from the Bousinesq hypothesis. These equations adopt a series of model constants, which have been evaluated for a variety of different flows. The default and strongly suggested constants are captured below in Tab.4.1.

TABLE 4.1. Standard $k - \epsilon$ model default constants

Constant	Value
C_μ	0.09
σ_k	1.00
σ_ϵ	1.30
$C_{\epsilon 1}$	1.44
$C_{\epsilon 2}$	1.92

Priyadarshini *et al.* altered the $C_{\epsilon 1}$ in their work, decreasing it to 1.40 [81]. As these constants are all directly modifiable in ANSYS Fluent, the same change has been made.

A strong assumption made by the standard $k - \epsilon$ model is that the flow is fully turbulent and the molecular viscosity is negligible. Whilst this assumption is acceptable for the general BKN flow field, it breaks down close to the wall in the near wall boundary flow. This is discussed in more depth later, however this assumption is addressed in the RNG model.

4.4.2 RNG $k - \epsilon$ Model

The RNG $k - \epsilon$ model is formulated using a statistical technique called renormilisation group theory, refining the standard $k - \epsilon$ model in a few ways.

- Rapidly strained flows are accounted for using an additional term in the ϵ equation (Eq.4.33).
- The effects of swirl are modelled.
- Prandtl numbers are defined using an analytical formula, rather than a set constant.
- The effective viscosity is calculated using a differential formula that accounts for viscous effects, addressing the aforementioned assumption of a fully turbulent flow.

The calculation of the effective viscosity is a worthwhile addition, as a conclusion drawn from literature was the importance of boundary flow treatment and its effects on wall measurements such as heat transfer. The differential equation takes the following form:

$$d\left(\frac{\rho^2 k}{\sqrt{\epsilon\mu}}\right) = 1.72\frac{\hat{v}}{\sqrt{\hat{v}^3 - 1 + C_v}}d\hat{v} \tag{4.35}$$

$$\hat{v} = \frac{\mu_{eff}}{\mu} \quad \text{and} \quad C_v \approx 100 \tag{4.36}$$

Eq.4.35 is then integrated, which in turn formulates a more accurate description on how the effective viscosity μ_{eff} deviates with the Reynolds number, a value that describes the degree of turbulence. $\alpha_k\mu_{eff}$ and $\alpha_\epsilon\mu_{eff}$ replace the $\mu + \frac{\mu_t}{\sigma_k}$ and $\mu + \frac{\mu_t}{\sigma_\epsilon}$ terms in Eq.4.32 and Eq.4.33 respectively. It is asserted in the theory guide that this technique should be paired with appropriate near-wall treatment, discussed in Sec.4.4.4 [49].

4.4.3 Generalised $k - \omega$ (GEKO) Model

It is an acknowledged problem by ANSYS that a single turbulence model cannot cover all flow types. CFD codes offer a different array of choices however ANSYS have implemented the GEKO model as a way to blend existing models into a single flexible choice. As it became clear through a survey of existing works that a comparison between models may be a necessary approach to conclusively decide the best choice for BKN, the GEKO model offered value to make this approach easier. It has a set of "free" parameters that are decoupled from the model constants, avoiding the loss of calibration that comes with altering their default values. These parameters have set values that emulate different models and can be tweaked as required for the application. They are summarised as follows:

- C_{SEP} - Optimises flow separation from smooth surfaces (0.7 - 2.5)
- C_{NW} - Optimises flow in non-equilibrium near wall regions (-2 - 2)
- C_{MIX} - Optimises strength of mixing in free shear flows (0 - 1)
- C_{JET} - Optimises free jets independent of the mixing layer (0 - 1)

This model is developed based upon the $k - \omega$ model, which isn't considered in this work. In order to emulate the standard $k - \epsilon$ model, the parameter values are 1, 1, 0, 0 respectively, with the added benefit of improved near-wall treatment that is native to the $k - \omega$ model. Strokach *et al.* did note that this parameter combination with an increase to the C_{MIX} value yielded strong conformity to experimental data in a GO2/CH_4 single injector combustion chamber [94]. It will thus be considered in this work as an additional source of comparison.

4.4.4 Wall Treatment

The overall fidelity of a CFD simulation is effected by the presence of walls. Referring back to Fig.4.2, there is a rigorous momentum transfer and harsh change in the velocity gradients with viscous forces dominating in the boundary flow. This in turn induces large changes in solution variables, thus an appropriate treatment of the wall is paramount to the accurate prediction of wall bounded turbulent flows such as BKN.

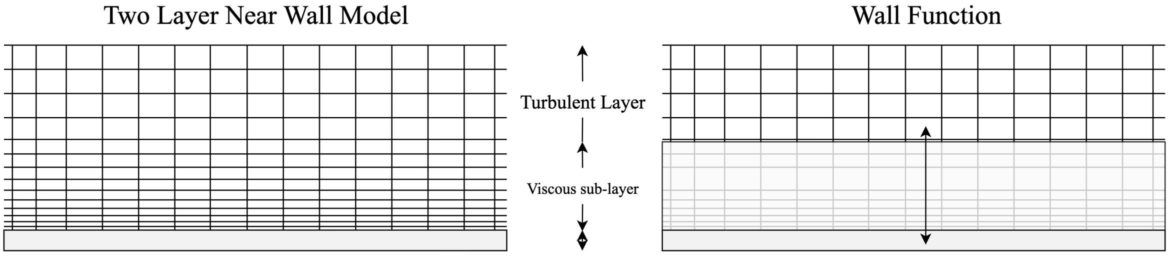


FIGURE 4.3. Two-layer near-wall modelling vs the wall function approach for boundary flows

A simple division can be made in the boundary layer, depicted in Fig.4.3. The turbulent layer maintains the $k-\epsilon$ assumption that the turbulent effects are dominant, rendering the molecular viscosity and other low Reynolds effects negligible. However, in the viscous sub-layer the laminar effects dominate, with the molecular viscosity responsible for momentum and heat transfer. An interim region between the two layers exists, where both turbulent and laminar effects are equally as important. Typically, two approaches that deal with this division exist; wall functions and a fully resolved two layer model. The wall function approach uses an empirical formula to bridge the viscous sub-layer without explicitly resolving it, suitable for coarse grids. However, with a fine mesh ($y^+ < 15$), wall functions deteriorate in accuracy, requiring grid insensitive models that resolve the flow all the way up to the wall. For ϵ equation models, ANSYS Fluent offers the enhanced wall treatment (EWT) and Menter Lechner options, seen in the work by Chemnitz [14]. It is also recommended by the ANSYS theory guide to adopt either of these approaches when using the $k-\epsilon$ model [49]. Fully resolved mesh insensitive models are native to the $k-\omega$ family and thus the GEKO model.

4.4.4.1 Enhanced Wall Treatment (EWT)

The enhanced wall treatment approach resolves the two layer boundary flow, with the demarcation of the regions governed by a wall distance based turbulent Reynolds number Re_y, defined in Eq.4.37 below. Here, y is the normalised wall distance calculated at the cell centers.

$$Re_y = \frac{\rho y \sqrt{k}}{\mu} \tag{4.37}$$

The threshold Reynolds number Re_y^* is 200, such that when $\mathrm{Re}_y > \mathrm{Re}_y^*$, the equations governing the $k - \epsilon$ model are employed. In the case that $\mathrm{Re}_y < \mathrm{Re}_y^*$, the one equation model of Wolfenshtein is used containing an alternate formulation for the turbulent viscosity [104]. A blending function is used to smoothly transition between the layers. A more detailed mathematical breakdown is seen in the Fluent theory guide [49].

4.4.4.2 Menter Lechner

The Menter Lechner formulation is an alternate way of modelling the boundary flow designed to address 2 key problems with the enhanced wall treatment option. Firstly, laminar flows with a Reynolds number below 200 may still exist in regions far from the wall, triggering the equations used to treat the viscous sub-layer where it isn't needed. Secondly, pressure gradients are also not handled in Wolfenshtein's equations, causing the layer switching location to alter the solution. When adopting the Menter Lechner approach, the ϵ equation is overwritten in the viscous sub-layer, utilising an algebraic function based on a simple mixing length model. Again, the mathematical breakdown can be found in the Fluent theory guide [49].

4.4.5 Additional Unclosed Terms

The Reynolds stress term $\widetilde{\bar{\rho} u_i'' u_j''}$ is dealt with using the turbulence modelling techniques discussed above. However in the set of Favre averaged Navier-Stokes equations Eq.4.26-4.29 several additional unclosed terms remain, including the species flux $\widetilde{\bar{\rho} u_i'' Y_k''}$, enthalpy flux $\widetilde{\bar{\rho} u_i'' h_s''}$ and the species chemical reaction rate $\overline{\dot{\omega}_k}$. When required, the same velocity gradient assumption seen in the Bousinesq hypothesis is used to close $\widetilde{\bar{\rho} u_i'' Y_k''}$ and $\widetilde{\bar{\rho} u_i'' h_s''}$ [80]. Alternatively, the PDF-flamelet method of modelling turbulent combustion removes the need to close these terms as the species equations are not solved and the energy equation is simplified. The same applies to $\overline{\dot{\omega}_k}$ as it is typically dealt with using a turbulent combustion model. Again the PDF-flamelet method removes the need for its calculation all together. Refer to Chapter 5 for a detailed breakdown of combustion modelling.

4.4.6 Turbulent Schmidt Number

The turbulent Schmidt number Sc_t, defined in Eq.4.38, compares the momentum and species diffusion (D_k), It is often paired with the Prandtl number Pr_t which compares momentum and thermal diffusion. The relation between the two values is defined using the Lewis number Le, seen in Eq.4.39.

$$Sc_t = \frac{\mu_t}{\rho D_t} \quad \text{and} \quad Pr_t = \frac{\mu_t C_p}{\lambda} \tag{4.38}$$

$$Sc_t = LePr_t \tag{4.39}$$

The default turbulent Schmidt number for the $k - \epsilon$ model in Fluent is 0.85, which is on the higher end of typical use cases. The average value seen in literature is 0.7, with many works lowering the value even further to ranges of 0.4-0.6 [84, 15, 52, 46]. When decreasing this value, the turbulent diffusion increases, which in turn enhances mixing and combustion. The type of flame simulated in this work is a non-premixed diffusion flame, and as the name suggests, the process of diffusion governs its behaviour. Thus, there is significant importance in selecting the correct value for the turbulent Schmidt number, however it also has the power to alter the flame shape unrealistically. Knowing this, a comparison was made, utilising turbulent Schmidt numbers of 0.55, 0.7 and 0.85 with respect to experimental data (see Chapter 7).

4.5 Equation of State

The correct selection of the EoS utilised in a CFD simulation is paramount for the overall accuracy and closure of the governing equations. The simplest form is referred to as the ideal gas law, which relates temperature, pressure and density, seen in Eq.4.40.

$$p = \rho \frac{R}{W} T \tag{4.40}$$

Here, R is the perfect gas constant and W is the molecular weight. It was well established through numerical works pertaining to similar conditions as BKN that this relation is inaccurate in the supercritical regime. It is also not applicable for liquids and solids. Once a fluid enters a supercritical state, the properties that explicitly differentiate the liquid and gas phases become less defined. The point at which this occurs for methane and oxygen is captured in Tab.3.2. Experimental works that explored the physical nature of supercritical flows, discussed in Sec.3.4.1, helped inform the development of "real gas" equations of state. Through an extensive survey, the Soave-Redlich-Kwong (SRK) cubic equation of state presented as the superior option. It is developed from the general form of a cubic equation of state, defined as:

$$P = \frac{RT}{V - b + c} - \frac{\alpha}{V^2 + \delta V + \epsilon} \tag{4.41}$$

Here, P is the absolute pressure (Pa), V is the specific molar volume (m^3/kmol) and T is the temperature (K). For the SRK equation of state the $\alpha, b, c, \delta, \epsilon$ coefficients are defined as follows:

$$\alpha(T) = \alpha_0 \left[1 + n \left(1 - \sqrt{\frac{T}{T_c}} \right) \right]^2 \quad \text{with} \quad n = 0.48 + 1.574\omega = 0.176\omega^2 \tag{4.42}$$

$$c = \epsilon = 0 \tag{4.43}$$

$$\alpha_0 = \frac{0.42747 R^2 T_c^2}{P_c} \tag{4.44}$$

$$\delta = b = \frac{0.08664 R T_c}{P_c} \tag{4.45}$$

The SRK EoS requires 3 parameters, the critical pressure p_c, the critical temperature t_c and the accentric factor ω. Fluent has these values predefined in its internal database. Note, these values are defined per species, therefore a multi-species mixture needs a method of combining these values based on its overall composition. The equations that govern this process are called mixing laws. Fluent houses multiple mixing laws, with the simplest method referred to as mass or mole fraction averaging. This involves taking a mass weighted sum of the individual species for the chosen parameter. In Eq.4.46 the mole weighted average of the accentric factor is given as an example. Subscript m is used to denote the mixture value.

$$\omega_m = \sum_{k=1}^{N} X_k \omega_k \tag{4.46}$$

Another method available is the one-fluid van der Waals mixing rule in which the SRK coefficients are calibrated as such:

$$\sqrt{\alpha_m} = \sum_{k=1}^{N} X_i \sqrt{\alpha_k} \tag{4.47}$$

$$b_m = \sum_{k=1}^{N} X_k b_k \tag{4.48}$$

These equations are then substituted into the original SRK to yield expressions for the mixture critical components as follows:

$$T_{cm} = \frac{\left[\sum_{k=1}^{N}\left(X_k \frac{T_{ck}}{P_{ck}^{0.5}}\right)\right]^2}{\sum_{k=1}^{N}\left(X_k \frac{T_{ck}}{P_{ck}}\right)} \tag{4.49}$$

$$P_{cm} = \frac{T_{cm}}{\sum_{k=1}^{N}\left(X_k \frac{T_{ck}}{P_{ck}}\right)} \tag{4.50}$$

$$V_{cm} = \sum_{k=1}^{N} \frac{X_k P_{ckV_{ck}}}{T_{ck}} \left(\frac{T_{cm}}{P_{cm}} \right) \tag{4.51}$$

Alongside the SRK EoS, the mixing laws for the critical parameters have been left as the Fluent default. The mole weighted mixing law is used for the accentric factor, with the remaining critical properties calculated using the one-fluid van der Waals mixing rule. The benefits of different choices have been not explored in this work, given that the expected return on accuracy is minimal.

4.5.1 Thermodynamic Properties

The thermodynamic properties of specific heat C_p, enthalpy h and entropy s are importable into ANSYS Fluent in Chemkin format. This format summarises the isobaric data over a given temperature range using piecewise polynomials. NASA polynomials are commonly used however they lack validity at cryogenic temperatures. The thermodynamic file used in this work employed self generated polynomial coefficients for this range, fitted from NIST data. This file was provided by Clara Morris, the previous student, with no additional modelling deemed worthwhile in the time-frame of this project [75]. It is important to note that the values computed from the thermodynamic file are considered "ideal state" at 1 standard atmosphere (1.01325 bar). ANSYS Fluent uses departure functions to adjust these values per the chosen EoS for the given operating conditions, which in BKN's case, are far from ideal.

4.5.2 Transport Properties

The transport properties of molecular viscosity and thermal conductivity can be defined for each individual species or for the mixture as a whole. In a highly turbulent flow these values are dominated by their turbulent counterparts, representing a key assumption of the standard $k - \epsilon$ model. With the additional effort to incorporate viscous effects where they are required, for example in the boundary flow, it is worthwhile modelling these values accurately. This is supported by Zhukov who noted the impacts on wall heat flux measurements [108]. On top of built in functions, Fluent allows the user to manually set the mixing method. This is done through the import of C code user defined functions (UDF's). On an individual species level, a UDF that defines the transport property as a function of temperature may also be used. This approach was chosen, with the NASA Chemical Equilibrium and Applications (CEA) database selected as the source for equation coefficients [96]. Viscosity μ and thermal conductivity κ, are defined as follows:

$$\left.\begin{matrix} ln\mu \\ \\ ln\kappa \end{matrix}\right\} = AlnT + \frac{B}{T} + \frac{C}{T^2} + D \tag{4.52}$$

The sets of coefficients provided in the CEA database are typically for 2 or 3 temperature ranges, 200-1000 K, 1000-5000 K and even 5000-15000 K. The cryogenic temperatures of the injected oxygen are thus not valid, therefore NIST data is used to incorporate a new cryogenic temperature range into the UDF. The UDF generation process is discussed in Chapter 6.

Initial simulations in this work used the mass weighted approach for the mixture viscosity and thermal conductivity. A key issue when using this method is the diminishing of lighter species such as H_2. A more realistic mixing law for the viscosity, suitable for reactive cases, is the rule defined by Wilke [101]. Here, a nested summation is present with K denoting the species in the outer sum and k denoting the species in the inner sum.

$$\mu = \sum_K \frac{V_K \mu_K}{\sum_k V_K \phi_{K,k}} \text{ where } \phi_{K,k} = \frac{1}{\sqrt{8}} \left(1 + \frac{Y_K}{Y_k}\right)^{-0.5} \left[1 + \left(\frac{\mu_K}{\mu_k}\right)^{0.5} \left(\frac{Y_k}{Y_K}\right)^{0.25}\right]^2 \tag{4.53}$$

The same approach was applied for thermal conductivity, with the more realistic mixing law defined by Zipperer and Herning. Here, the same species subscript convention is used [109].

$$\kappa = \sum_K \frac{V_K \kappa_K}{\phi_K} \text{ where } \phi_K = \sum_K V_k \sqrt{\frac{Y_K}{Y_k}} \tag{4.54}$$

4.6 Computational Domain

4.6.1 Meshing and Discretisation

In order to apply the mathematical modelling developed in the previous section, the computational domain needs to be defined in order to facilitate the numerics. Looking back at Sec.3.4.2, the overall numerical process was broken down into 3 key steps, with the first step including the domain setup. This is done in the form of a grid/mesh, which breaks up the flow field into a series of cells, on which the calculations take place. Grid generation is an important step, as it can have a huge influence on the simulation fidelity. When a CFD calculation is executed, the integrated terms in the governing equations are replaced with discretised algebraic expressions, which are solved yielding individual values at each cell in the mesh. The division of BKN's flow field into the individual control volumes (cells) across a 2D axisymmetric structure was completed by Morris, as seen in Fig.4.4 [75]. Here, the boxed regions indicate key regions of interest with box 1 enclosing the fine resolution within the injector recess. Box 2 marks the expansion region in which the flame is expected to expand with box 3 marking the beginning of a radially uniform grid that spans down the chamber.

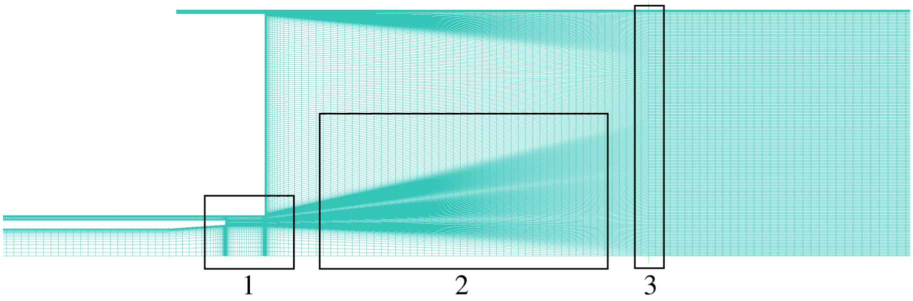


FIGURE 4.4. Simulation mesh with key areas highlighted, taken from [75]

Seen in Fig.4.3, the meshing resolution becomes finer at the wall in order to calculate the boundary flow and resolve the viscous sublayer. A dimensionless normal wall distance of $y^+ < 1$ was achieved. This is the threshold defined by Fluent before additional modelling is employed through the y^+ insensitive EWT approach. Fluent calculates this value based upon the turbulent kinetic energy, which is often referred to as y^* instead of y^+, given in Eq.4.55.

$$y^{*} = \frac{\rho y \sqrt{\sqrt{C_u} k}}{\mu} \tag{4.55}$$

The governing equations which have an infinite continuum of values throughout any domain, now have a finite number of volumes within the domain defined by the grid. The process of discretisation used by Fluent that allows the flow variables to be solved at these points is called the finite volume method. Within this category, the coupled pressure solver has been utilised which solves the mass and momentum equations simultaneously based upon a pressure correction. In this way an intertwined process occurs, by which the pressure corrected velocity field satisfies the mass continuity. Remaining equations such as energy, species and turbulence are solved separately. Each individual control volume is integrated over, a process that yields discrete algebraic expression for the variable at the cell centre and surrounding cells. The process of upwinding computes fluxes for the cell faces based off their centroids, with second order accuracy employing a taylor series expansion to execute this. For more detail, refer the to the Fluent theory guide [49].

4.6.2 Boundary Conditions

With the geometry setup, a set of boundary conditions needs to be defined. The governing equations can be applied to many flow calculations, however the boundary conditions are what differentiate individual solutions from one another. For BKN, the chamber walls, injection parameters and outlet represent the perimeters of the flow, influencing the housed behaviour. The comprehensive survey of physical experiments seen in Chapter 3 noted the impact of various injection conditions, with the BKN experiment testing two different load points [67]. Measurements of temperature, pressure and heat flux were all made at the wall, thus numerical validation requires a proper definition of the wall boundary. How the boundary conditions have been modelled is detailed in depth in Chapter 6.

CHAPTER 5

Case Study 2 - Combustion

This chapter presents the methods used to model combustion in this work. It captures the course content of MECH5265 with relevance to the simulation of BKN. An introduction to combustion chemical kinetics is seen in Section 5.1, followed by an overview of chemical mechanisms in Section 5.2. Section 5.3 introduces diffusion flames with Section 5.4 and 5.5 detailing their modelling within non-premixed turbulent combustion regimes.

5.1 Chemical Kinetics

It is important to note the basic chemical kinetics governing the combustion reaction before exploring its interplay with turbulent flow. Laidler defines chemical kinetics as the field that deals with the rates of chemical reactions and how the rates depend on factors such as concentration and temperature [55]. Eq.5.1 considers a chemical system with N species reacting through M reactions, in which the equilibrium state is signified with the $\rightleftharpoons$ symbol.

$$\sum_{k=1}^{N} v_{kj}^{'} \mathcal{M}_k \rightleftharpoons \sum_{k=1}^{N} v_{kj}^{''} \mathcal{M}_k \quad \text{for} \quad j = 1, 2, ..., M \tag{5.1}$$

Here, $\mathcal{M}_k$ is species k with $v_{kj}^{'}$ and $v_{kj}^{''}$ being the stoichiometric coefficients of species k in reaction j. The mass conservation equation (Eq.4.8) enforces:

$$\sum_{k=1}^{N} v_{kj}^{'} W_k = \sum_{k=1}^{N} v_{kj}^{''} W_k \quad \text{or} \quad \sum_{k=1}^{N} (v_{kj}^{''} - v_{kj}^{'}) W_k = 0 \quad \text{for} \quad j = 1, 2, ..., M \tag{5.2}$$

The mass reaction rate $\dot{\omega}$ which was introduced in Eq.4.11 can be defined for each individual species k as follows:

$$\dot{\omega}_k = \sum_{j=1}^{M} \dot{\omega}_{kj} = W_k \sum_{j=1}^{M} (v_{kj}^{''} - v_{kj}^{'}) Q_j \tag{5.3}$$

Summing all reaction rates ω_k is also 0, enforced by the conservation of mass. In Eq.5.3, the term Q_j is known as the progress rate of reaction j. It can be formally defined in terms of the forward reaction rate K_{fj} and the reverse reaction rate K_{rj}. It is a standard convention that the kinetic rates are expressed using molar concentrations X_k as seen in Eq.5.4.

$$Q_j = K_{fj} \prod_{k=1}^{N} X_k^{v_{kj}^{'}} - K_{rj} \prod_{k=1}^{N} X_k^{v_{kj}^{''}} \tag{5.4}$$

The reaction rates are modelled by the empirical Arrhenius law, defined in Eq.5.5. The law is broken up into a pre-exponential term A_{fj}, a temperature exponent β_j and the activation energy E_j. The final term is marked as the collision fraction, as it represents the fraction of chemical collisions that contain the energy to activate the reaction j.

$$K_{fj} = A_{fj} T^{\beta_j} \quad \underbrace{e^{-\frac{E_j}{RT}}}_{\text{Collision fraction}} \tag{5.5}$$

K_{rj} is calculated from K_{fj} using the change in enthalpy Δh and entropy Δs as follows:

$$K_{rj} = \frac{K_{fj}}{\left(\frac{p_a}{RT}\right)^{\sum_{k=1}^{N}(v_{kj}^{''} - v_{kj}^{'})} \exp\left(\frac{\Delta s_j}{R} - \frac{\Delta h_j}{RT}\right)} \tag{5.6}$$

Here, p_a is 1 bar. The three parameters A_{fj}, β_j and E_j required to calculate the reaction rates via the Arrhenius law are captured within the chemical mechanism, a kinetic scheme that details all elementary reactions in the combustion reaction.

5.2 Chemical Mechanism

Chemical mechanisms are one of the datasets required by the CFD solver during calculation and in the pre-processing stage. Fluent requires the mechanism to be given in standard CHEMKIN format. The elements are stated, followed by the species, followed by a detailed list of sequential reactions. The first column captures A_{fj} in cgs units, followed by the dimensionless β_j with the third column containing E_j in cal/mole. A snippet of the reduced DLR LOX/CH4 mechanism is given as an example below, as it is the primary scheme utilised in this work. The justification for its reduction is provided by Zhukov and Kong [107]. It was derived based on the RAMEC mechanism, a scheme developed for high pressure methane/air combustion in ramjet engines. Instead, the complete kinetics under rocket engine conditions using undiluted fuel-oxygen mixtures have been incorporated. The final scheme has been reduced to 5 elements, 23 species and 51 reactions.

Snippet of the Reduced LOX/CH4 DLR Mechanism

```
ELEMENTS
H C O N AR
END
SPECIES
H2      H       O       O2      OH      H2O     HO2     H2O2
CH2     CH3     CH4     CO      CO2     HCO     CH2O    CH3O
C2H3    C2H4    C2H5    C2H6    CH3O2   N2      AR
END
REACTIONS      CAL/MOLE
O2+CH2O<=>HO2+HCO                        1.000E+14    0.000  40000.00
H+O2+M<=>HO2+M                           2.800E+18   -0.860      0.00
---------------------------------------------------------------------
HO2+CH3<=>O2+CH4                         1.000E+12    0.000      0.00
HO2+CH3<=>OH+CH3O                        2.000E+13    0.000      0.00
HO2+CO<=>OH+CO2                          1.500E+14    0.000  23600.00
HO2+CH2O<=>HCO+H2O2                      1.000E+12    0.000   8000.00
```

5.3 Diffusion Flames

5.3.1 Mixture Fraction Theory

Diffusion flames are the product of a combustion reaction in which the fuel and oxidiser are introduced into the chamber in separate streams. Under certain assumptions (discussed throughout), the chemistry is governed by mixture fraction theory, in which a single conserved scalar quantity $\mathcal{Z}$ is related to the thermochemical state of the fluid. $\mathcal{Z}$ is referred to as the mixture fraction which measures the local fuel and oxidiser ratio, normalised to fit between the range of 0-1. Pure fuel yields a value of 1, with pure oxidiser equal to 0.

$$\mathcal{Z} = \frac{sY_F - Y_O + Y_O^0}{sY_F^0 + Y_O^0} \tag{5.7}$$

Here s is the stoichiometric ratio with subscript O and F indicating oxidiser and fuel mass fractions respectively. The inlet mass fractions are marked with superscript 0. Eq.5.7 defines the mixture fraction in a one step reaction, thus a definition which accounts for multiple species and reactions is required, seen in Eq.5.8. Here, the mixture fraction is constructed using the mass fractions of reaction elements Z_p where a_{kp} is the number of elements of type p in species k.

$$\mathcal{Z} = \frac{Z_p - Z_p^O}{Z_p^F - Z_p^O} \quad \text{where} \quad Z_p = W_p \sum_{k=1}^{N} a_{kp} \frac{Y_k}{W_k} \tag{5.8}$$

Two core assumptions used in this derivation include:

- The diffusion coefficients for all species are equal ($D_k = \mathcal{D}$).
- The Lewis number (introduced in Sec.4.4.6) is equal to 1.

$$Le = \frac{\kappa}{\rho C_p \mathcal{D}} = \frac{Pr_t}{Sc_t} = 1 \tag{5.9}$$

Both these assumptions are deemed acceptable for turbulent flows [49]. $\mathcal{Z}$ remains unique under the assumption that the diffusion coefficients for all species are equal. Furthermore, assumptions surrounding the structure of the diffusion flame can be made, which in turn allows primitive variables such as species mass fractions and temperature to expressed solely as a function of $\mathcal{Z}$.

5.3.2 Steady Strained Counter-flow Diffusion Flame

The diffusion flame structure considered in this work is known as the steady strained counter-flow flame. Opposing jets of fuel and oxidiser yield a flame front at the point of stoichiometry, seen in Fig.5.1. This type of structure is viewed locally as one-dimensional, dependant on the coordinate normal to the flame front. The normal spatial coordinate x, can be mapped to the mixture fraction $\mathcal{Z}$ which varies from 0-1 from the oxidiser inlet to the fuel inlet. The constant strain on the flame implies that it is steady, thus time is not a consideration in the transformation.

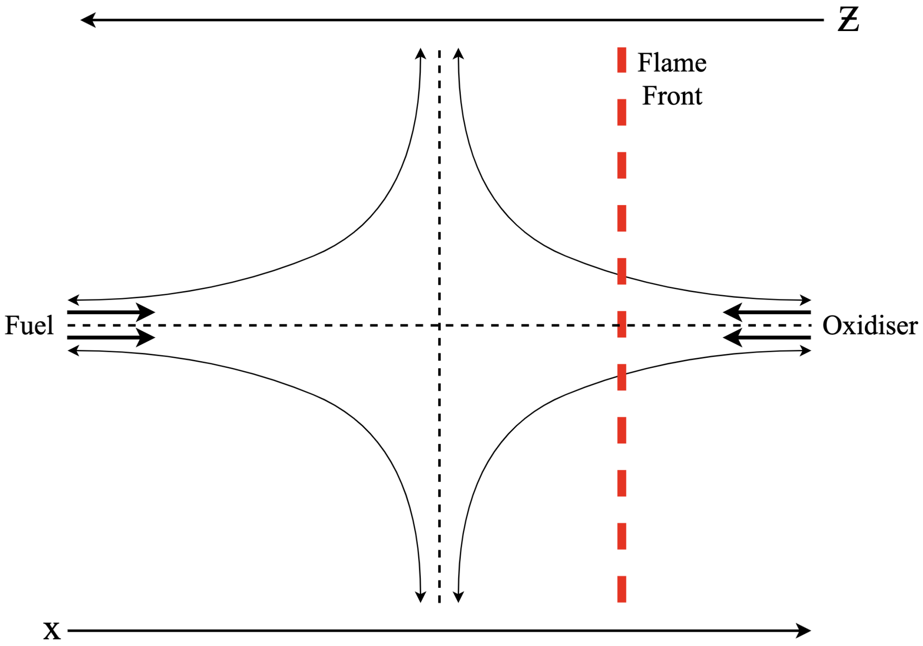


FIGURE 5.1. Graphical illustration of a counterflow diffusion flame

As mentioned, a coordinate substitution can be made for the x coordinate, transforming the system to mixture fraction space. As the flame is assumed to be steady, the mixture fraction depends only on the spatial coordinate x, and not time. This relationship is defined in Eq.5.10, derived in [80].

$$\mathcal{Z} = \frac{1}{2}\left(1 - \text{erf}\left(x\sqrt{\frac{a}{2\mathcal{D}}}\right)\right) \tag{5.10}$$

Here, a is the strain rate, dictated by the velocity and distance between the jets, with an increasingly strained flame eventually resulting in extinction. $\mathcal{D}$ references the assumption of equal diffusivities, acting as the global mass diffusivity coefficient. With the physical space mapped to the mixture fraction, the same can be done for the strain rate via a new variable called the scalar dissipation $\mathcal{X}$. It has a dimension of inverse time, and can be thought of as a measure of mixture fraction gradients.

$$\mathcal{X} = 2\mathcal{D}\left(\frac{\partial \mathcal{Z}}{\partial x_i}\frac{\partial \mathcal{Z}}{\partial x_i}\right) \tag{5.11}$$

With these two parameters defined, the original species conservation equations can now be expressed using the mixture fraction and scalar dissipation as follows:

$$\rho\frac{\partial Y_k}{\partial t} = \dot{\omega_k} + \frac{1}{2}\rho\mathcal{X}\frac{\partial^2 Y_k}{\partial \mathcal{Z}^2} \tag{5.12}$$

Additionally, temperature can be expressed in a similar manner:

$$\rho\frac{\partial T}{\partial t} = \frac{1}{2}\rho\mathcal{X}\frac{\partial^2 T}{\partial \mathcal{Z}^2} - \frac{1}{C_p}\sum_{k=1}^{N} h_k\dot{\omega_k} + \frac{1}{2C_p}\rho\mathcal{X}\left[\frac{\partial C_p}{\partial \mathcal{Z}} + \sum_{k=1}^{N} C_{pk}\frac{\partial Y_k}{\partial \mathcal{Z}}\right]\frac{\partial T}{\partial \mathcal{Z}} \tag{5.13}$$

Eq.5.12 and Eq.5.13 are known as the flamelet equations. Once the scalar dissipation is defined, the flamelet equations can be solved in $\mathcal{Z}$ space. As the flame discussed here is steady, temperature and species mass fractions become a function of the mixture fraction only. As a result, the temporal derivatives in Eq.5.12 and Eq.5.13 are equal to 0. Both equations contain the species reaction rate term ω_k, with its value calculated through the forward and reverse reaction rates discussed previously.

5.4 PDF-Flamelet Method

As previously mentioned in Sec.4.4.5, Eq.4.28 contains the mean reaction rate term $\bar{\dot{\omega}}$, a term that is dealt with using a turbulent combustion model. When breaking down this term, it cannot easily be expressed as a function of other mean quantities such as mass fractions, density and temperature. An alternative approach involves physical analysis, rather than a mathematical expansion and relation to other mean variables. The three-dimensional combustion flame front can be assumed to be composed of multiple counterflow diffusion flames, called flamelets. The geometrical descriptions assume a large Damköhler number, the ratio between chemical time scales and turbulent time scales. With a large, but not infinitely large Damköhler number, the reaction time is much shorter than the turbulent mixing process. Physically, this implies the flame front in which the chemical reactions occur is thin, viewed locally as one-dimensional. Hence, the turbulent flame brush is modelled as an ensemble of discrete diffusion flamelets, depicted in Fig.5.2. This analysis underpins the PDF-flamelet method, a computationally efficient approach that omits the requirement to calculate individual species equations (Eq.4.28). Instead, two mixture fraction transport equations are solved for, avoiding the need to provide closure for the mean reaction rate term $\bar{\dot{\omega}}$.

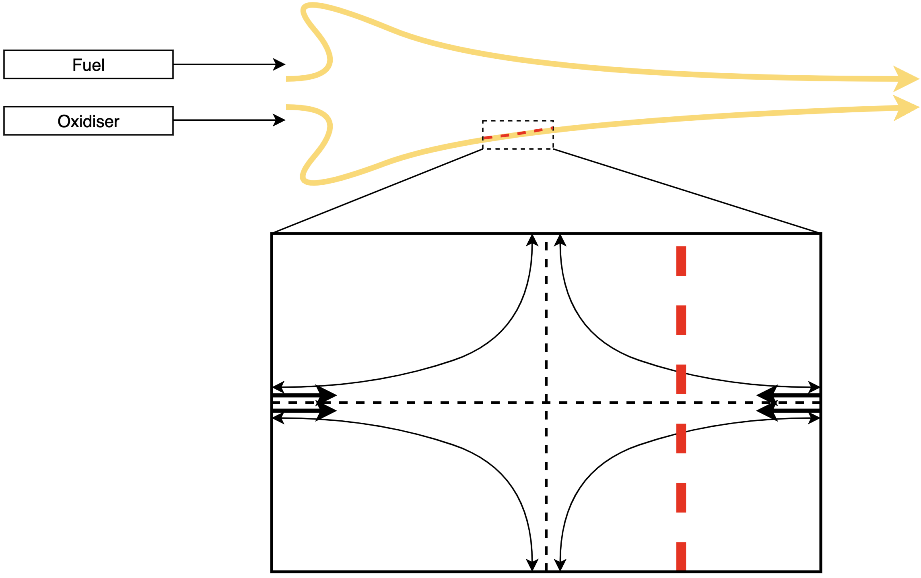


FIGURE 5.2. Geometric analysis of the non-premixed instantaneous flame front

Within the context of a RANS simulation, the process can be divided into five steps. Note, the following procedure is tailored to the method used by ANSYS Fluent.

(1) The flame structure is defined by the flamelet equations (Eq.5.12 and Eq.5.13), with the species mass fraction and temperature calculated as functions of the mixture fraction Z and stoichiometric scalar dissipation $\mathcal{X}_{st}$.

(2) The results are tabulated in flamelet libraries, with each table in the library containing a set of temperatures and species mass fractions over a range of distinct mixture fraction values between 0 and 1. Each table is parameterised by a single stoichiometric scalar dissipation value.

(3) A lookup table containing time averaged parameters is formed using assumed probability density functions (PDF's). The flamelet functions are combined with the PDF's and are integrated over the range of mixture fraction and stoichiometric scalar dissipation values. This produces a final table containing functions for the mean values of temperature $\widetilde{T}$, species mass fraction $\widetilde{Y_k}$ and density $\widetilde{\rho}$, defined below:
    - $\widetilde{T}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}, \widetilde{\mathcal{X}_{st}})$
    - $\widetilde{Y_k}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{\mathcal{X}_{st}})$
    - $\bar{\rho}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}, \widetilde{\mathcal{X}_{st}})$

(4) During the RANS calculations, the mean mixture fraction $\widetilde{Z}$, mean mixture fraction variance $\widetilde{Z}''^2$, mean enthalpy $\widetilde{H}$ and mean stoichiometric scalar dissipation $\widetilde{X}_{st}$ are solved for. Using these values, the PDF table is referenced to obtain $\widetilde{T}$, $\widetilde{Y_k}$ and $\widetilde{\rho}$.

(5) The density term is then corrected to account for real gas effects using the SRK EoS discussed in Sec.4.5.

This procedure can also be referred to as the primitive variable method as it pre-tabulates temperature, species mass fraction and density. This is what allows the reduction the governing set of differential equations, as no additional species equations (Eq.4.28) are required. The DLR Zhukov/Kong mechanism contains 23 species, hence 22 equations are omitted and replaced by two mixture fraction equations (discussed below). In combination with the pre-processing of these variables, an immense saving in computational time can be achieved. These overarching steps are broken down in following subsections.

5.4.1 Flamelet Calculations

Steady diffusion flamelets do not require a consideration of time, thus Eq.5.12 and Eq.5.13 can be re-written as functions of $\mathcal{Z}$ and $\mathcal{X}$ only.

$$\dot{\omega_k} + \frac{1}{2}\rho\mathcal{X}\frac{\partial^2 Y_k}{\partial \mathcal{Z}^2} = 0 \tag{5.14}$$

$$\frac{1}{2}\rho\mathcal{X}\frac{\partial^2 T}{\partial \mathcal{Z}^2} - \frac{1}{C_p}\sum_{k=1}^{N} h_k\dot{\omega_k} + \frac{1}{2C_p}\rho\mathcal{X}\left[\frac{\partial C_p}{\partial \mathcal{Z}} + \sum_{k=1}^{N} C_{pk}\frac{\partial Y_k}{\partial \mathcal{Z}}\right]\frac{\partial T}{\partial \mathcal{Z}} = 0 \tag{5.15}$$

The instantaneous species reaction rate ω_k is calculated via the Arrhenius law, detailed in section 5.1. C_{pk} denotes the specific heat for each species k. A mass averaging law is used to calculate C_p, the specific heat for the entire mixture. The specific enthalpy for each species h_k is calculated using C_p via Eq.5.30. Density ρ is calculated from ideal gas law based on the specified operating pressure. Finally, two important inputs are provided to Fluent. First is the table resolution, a parameter that discretises the mixture fraction continuum between 0 and 1. The distribution of individual mixture fraction values is concentrated around the stoichiometric point, approximately 0.2 for LOX/CH$_4$. The second is the stoichiometric scalar dissipation $\mathcal{X}_{st}$, a representation of the aerodynamic strain and deviation from chemical equilibrium. The high Damköhler number assumption, corresponding to the thin flame front, allows the flamelet to be evaluated around the point of stiochiometry, marked by the red line in Fig.5.2. Each $\mathcal{X}_{st}$ value parameterises the range of mixture fraction values, with its relationship to the general scalar dissipation (required in Eq.5.14 and Eq.5.15) given in Eq.5.16 and Eq.5.17. Here, ρ_∞ is the density in the oxidiser stream.

$$\phi(\mathcal{Z},\rho) = \frac{a_s}{4\pi}\frac{3\left(\sqrt{\frac{\rho_\infty}{\rho}}+1\right)^2}{2\sqrt{\frac{\rho_\infty}{\rho}}+1}e^{-2(\mathrm{erfc}^{-1}(2\mathcal{Z}))^2} \tag{5.16}$$

$$\mathcal{X}(\mathcal{Z}) = \mathcal{X}_{st}\frac{\phi(\mathcal{Z},\rho)}{\phi(\mathcal{Z}_{st},\rho_{st})} \tag{5.17}$$

5.4.2 Flamelet Tabulation

With the calculations executed, the results are tabulated in a flamelet text file. Each table is characterised by 5 distinct values, captured in the headder. This includes the stoichiometric scalar dissipation $\mathcal{X}_{st}$, the number of species, the table resolution (gridpoints), stoichiometric mixture fraction $\mathcal{Z}_{st}$ and the pressure. The table itself contains the mixture fractions values ranging from 0-1, the corresponding temperature values and the corresponding mass fractions of each species. Typically the first table has a stoichiometric scalar dissipation value close to 0, which is then increased until the extinction point is reached. This captures the increasing strain on the flame as it departs from chemical equilibrium.

5.4.3 PDF Tabulation

So far the flamelet tables capture counter-flow diffusion flame values for temperature and species mass fraction independent of the turbulent flow field. In the context of a RANS simulation, averaged values are required which can be sourced by the CFD solver. This is done using probability density functions (PDF's). PDF's can be thought of as the fraction of time that the fluid spends in vicinity of the state, depicted in Fig.5.3. Here the integral of the range denoted by $\Delta\mathcal{Z}$ corresponds to the probability of $\mathcal{Z}$ taking a value in this range within the time scale t. The profiles for flamelet quantities are convoluted with PDF's and integrated over in order to calculate mean values for temperature and species mass fraction, defined in Eq.5.18 and Eq.5.19. Mean density can also be calculated in a similar manner.

$$\widetilde{Y_k} = \int \int_0^1 Y_k(\mathcal{Z}, \mathcal{X}_{st}) p(\mathcal{Z}, \mathcal{X}_{st}) d\mathcal{Z} d\mathcal{X}_{st} \tag{5.18}$$

$$\widetilde{T_k} = \int \int_0^1 T(\mathcal{Z}, \mathcal{X}_{st}, h) p(\mathcal{Z}, \mathcal{X}_{st}, h) d\mathcal{Z} d\mathcal{X}_{st} \tag{5.19}$$

$$\frac{1}{\bar{\rho}} = \int \int_0^1 \frac{\rho(\mathcal{Z}, \mathcal{X}_{st}, h)}{p(\mathcal{Z}, \mathcal{X}_{st}, h)} d\mathcal{Z} d\mathcal{X}_{st} \tag{5.20}$$

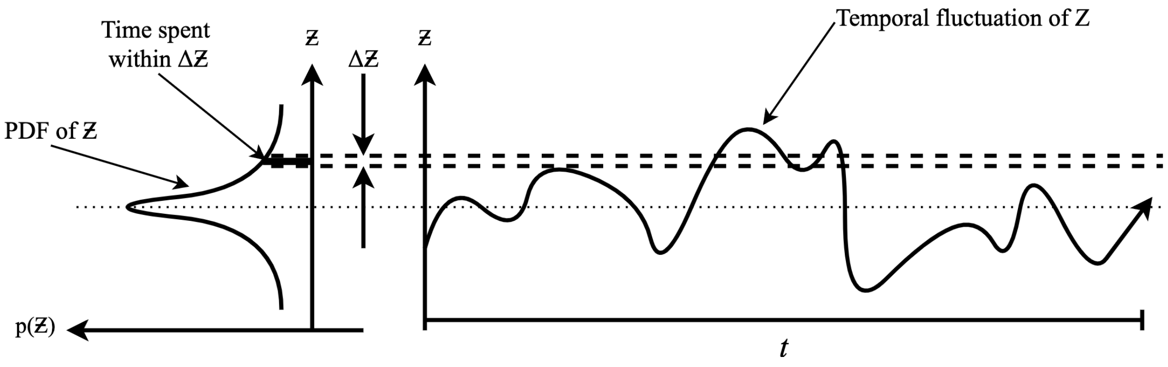


FIGURE 5.3. Example PDF function of $\mathcal{Z}$ as it fluctuates through time

Note that in Eq.5.19 and Eq.5.20, enthalpy h is included. This is due to the non-adiabatic treatment of the simulation, in which heat transfer is modelled through the chamber walls. This necessitates the inclusion of enthalpy, however it is included in a simplified manner. Heat gain/loss is assumed to have negligible impact on the species mass fractions, hence its omission in Eq.5.18.

Assuming the PDF's are statistically independent, the terms in Eq.5.18-Eq.5.20 can be broken down as such:

$$p(\mathcal{Z}, \mathcal{X}_{st}) = p(\mathcal{Z})p(\mathcal{X}_{st}) \quad \text{and} \quad p(\mathcal{Z}, \mathcal{X}_{st}, h) = p(\mathcal{Z})p(\mathcal{X}_{st})p(h) \tag{5.21}$$

The shape of the mixture fraction PDF is assumed to follow β functions, defined as follows:

$$p(\mathcal{Z}) = \frac{\mathcal{Z}^{\alpha-1}(1-\mathcal{Z})^{\beta-1}}{\int \mathcal{Z}^{\alpha-1}(1-\mathcal{Z})^{\beta-1} d\mathcal{Z}} \tag{5.22}$$

$$\alpha = \widetilde{\mathcal{Z}} \left[\frac{\widetilde{\mathcal{Z}}(1-\widetilde{\mathcal{Z}})}{\widetilde{\mathcal{Z}''^2}} - 1 \right] \quad \text{and} \quad \beta = (1-\widetilde{\mathcal{Z}}) \left[\frac{\widetilde{\mathcal{Z}}(1-\widetilde{\mathcal{Z}})}{\widetilde{\mathcal{Z}''^2}} - 1 \right] \tag{5.23}$$

Here, the Favre averaged mixture fraction $\mathcal{Z}$ and mixture fraction variance $\mathcal{Z}''^2$ are required. These variables are computed during the flow field calculations, discussed in Sec.5.4.4.

Fluent assumes the scalar dissipation is unchanged along the flame front, thus a simple Direc-delta function is employed for the PDF. A similar assumption is applied to enthalpy, with the heat losses deemed to have a negligible impact on the turbulent enthalpy fluctuations. Under this assumption it can also be defined using a Direc-delta function as follows:

$$p(\mathcal{X}_{st}) = \delta(\mathcal{X}_{st} - \widetilde{\mathcal{X}_{st}}) \quad \text{and} \quad p(h) = \delta(h - \widetilde{h}) \tag{5.24}$$

Here, the Favre averaged stoichiometric scalar dissipation $\widetilde{\mathcal{X}_{st}}$ and enthalpy $\widetilde{h}$ are required. Again, these values are computed during the flow field calculations.

Similarly to the flamelet tables, Fluent pre-calculates the mean values of temperature, species mass fraction and density and stores them in a lookup table for reference during the flow field calculation. The required inputs are described below in Eq.5.25 with the computation detailed below in Sec.5.4.4.

$$\widetilde{T}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}, \widetilde{\mathcal{X}_{st}}) \qquad \widetilde{Y_k}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{\mathcal{X}_{st}}) \qquad \bar{\rho}(\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}, \widetilde{\mathcal{X}_{st}}) \tag{5.25}$$

5.4.4 Flow field Calculations

The pre-processing stage covers the steps described up until this point when using the PDF-flamelet model. During the numerical calculations, it was previously mentioned that the set of Favre averaged species equations (Eq.4.28) are no longer need to be solved as the mass fractions are stored in the PDF lookup table. Instead values for $\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}$ and $\widetilde{\mathcal{X}_{st}}$ are required, in order to extract $\widetilde{T}, \widetilde{Y_k}$ and $\bar{\rho}$. In place of the species equations, only two additional transport equations are required to fulfill these requirements. They are defined as follows:

$$\frac{\partial(\bar{\rho}\widetilde{\mathcal{Z}})}{\partial t} + \frac{\partial}{\partial x_i}(\bar{\rho}u_i\widetilde{\mathcal{Z}}) = \frac{\partial}{\partial x_i}\left[\left(\frac{\kappa}{C_p} + \frac{\mu_t}{\sigma_t}\right)\frac{\partial\widetilde{\mathcal{Z}}}{\partial x_i}\right] \tag{5.26}$$

$$\frac{\partial(\bar{\rho}\widetilde{\mathcal{Z}''^2})}{\partial t} + \frac{\partial}{\partial x_i}(\bar{\rho}u_i\widetilde{\mathcal{Z}''^2}) = \frac{\partial}{\partial x_i}\left[\left(\frac{\kappa}{C_p} + \frac{\mu_t}{Pr_t}\right)\frac{\partial\widetilde{\mathcal{Z}''^2}}{\partial x_i}\right] + C_g\mu_t\left(\frac{\partial\widetilde{Z}}{\partial x_i}\right)^2 - C_d\bar{\rho}\frac{\epsilon}{k}\widetilde{\mathcal{Z}''^2} \tag{5.27}$$

Here, κ is the laminar thermal conductivity, computed using the mixing law defined in Eq.4.54. C_g and C_d are constants defined as 2.86 and 2.0 respectively. Pr_t is the turbulent Prandtl number, introduced in comparison to the turbulent Schmidt number in Eq.4.38. The turbulent Schmidt number is not required for each individual species and thus is re-named by Fluent to the PDF-Schmidt number. The turbulent Prandlt number Pr_t is set to equal the PDF Schmidt number, enforced by the assumption that the Lewis number is equal to 1.

$$Sc_{tPDF} = LePr_t \quad \text{where} \quad Le = \frac{\kappa}{\rho C_p \mathcal{D}} = 1 \tag{5.28}$$

This assumption simplifies the energy equation (Eq.4.29), which is solved for the mean enthalpy term $\widetilde{h}$.

$$\frac{\partial}{\partial t}(\bar{\rho}\widetilde{h}) + \frac{\partial}{\partial x_i}(\bar{\rho}\widetilde{u_i}\widetilde{h}) = \frac{\partial}{\partial x_i}\left(\frac{\kappa_{eff}}{C_p}\frac{\widetilde{h}}{\partial x_i}\right) \tag{5.29}$$

Here, κ_{eff} is the sum of the laminar and turbulent thermal conductivity's. The total enthalpy h is calculated using a mass weighted mixing law, with the individual enthalpies calculated as follows:

$$h = \sum_{j=1}^{M} Y_j h_j \quad \text{where} \quad h_j = \int_{298.15}^{T} C_{pk} Dt + h^0_{fk}(298.15) \tag{5.30}$$

Finally, the mean stoichiometric scalar dissipation needs to be evaluated. Fluent models this term as being proportional to the turbulent time scale, defined using the turbulent kinetic energy k and its dissipation rate ϵ. This relation is defined in Eq.5.31, with $C_{\mathcal{X}}$ equal to 2.

$$\widetilde{\mathcal{X}_{st}} = C_{\mathcal{X}} \widetilde{\mathcal{Z}''^2} \frac{k}{\epsilon} \tag{5.31}$$

With the methods of evaluating $\widetilde{\mathcal{Z}}, \widetilde{\mathcal{Z}''}, \widetilde{h}$ and $\widetilde{\mathcal{X}_{st}}$ during the flow field calculations defined, the requirements for embedding the effects of combustion within the turbulent flow field via the PDF-flamelet method are complete. Alongside the presented equations, the turbulent transport equations (Eq.4.32 and Eq.4.33) are being solved, as well as the mass continuity (Eq.4.26) and momentum equations (Eq.4.27) for a complete computation of the flow field.

5.4.5 Limitations of the PDF-Flamelet Model

The assumptions and simplifications adopted by the PDF-flamelet method warrant certain limitations with its application. The high Damköhler number assumption reduces the scope of chemical reaction times, as the flame front is considered thin in which the reaction time is almost instantaneous. Consequently, the overproducion of species produced via slow reaction pathways occurs. Flame quenching is also not modelled. Individual species molecular diffusion is not captured due to assumption of a single diffusion coefficient. Within the flamelet equations, the mixture fraction boundaries are set by the oxidiser inlet and fuel inlet only. As mentioned in Chapter 3, Wu *et. al.* introduced a thermal wall boundary condition within mixture fraction space to incorporate flame-wall interaction, aimed to address underestimated wall heat flux [61]. These boundary conditions can be extended to include additional processes not captured by the PDF-flamelet method such as wall recombination reactions.

5.5 Eddy Dissipation Concept

The PDF-flamelet method boasts immense computational savings, however there is a loss in accuracy when compared to models such as the EDC. The EDC requires each species equation to be solved during the flow field calculation and thus the closure of the mean reaction rate term. The same high Damköhler assumption used by the PDF-flamelet method is utilised within a non-premixed combustion regime. The turbulence slowly mixes the fuel and oxisider into the reaction zone where they burn quickly in comparison. The EDC model applies this concept to relate the mean reaction rate term to the turbulent mixing time scale, devised by Magnussen [63]. The application of the EDC model was deemed a low priority in this work, thus it is not expanded on further. It was tested briefly however the computational time was not feasible given the resources available and the timeframe of the project.

CHAPTER 6

Methodology

This chapter will break down the research design and methods used to simulate BKN. Sec.6.1 presents the identified areas of investigation, informed by literature and the work completed by Morris [75]. Sec.6.2 discusses the methods used to model combustion, in particular the evaluation of PDF-flamelet parameters and different chemical mechanisms. Sec.6.3 introduces how the wall boundary conditions were setup for heat transfer analysis. Sec.6.4 details the integration of custom transport modelling, with Sec.6.5 outlining the various approaches used to model turbulence. Finally, Sec.6.6 outlines additional elements of the simulation setup, and the roadmap followed throughout the thesis timeline.

6.1 Areas of Investigation

The foundation for the approach taken in this thesis is two-fold; the existing literature pertaining to simulation conditions synonymous with BKN and the work completed by Morris 6 months prior. She constructed a 2D axisymmetric structured mesh, resolved to $y^+ < 1$ at the wall, with 168 187 cells in total. Fig.6.1 summarises the different sections in the BKN geometry. Using ANSYS Fluent, a successful simulation that synthesised 4 core components together was achieved, summarised in Fig.6.2. A custom thermodynamic file was also utilised, which extended low temperature ranges for oxygen, methane, carbon monoxide and carbon dioxide. Some additional parameters were tweaked, however many facets of the simulation were left default/simplified, opening up potential areas of exploration. As a result, six major categories were chosen as a priority for the project, with the intention of combining the net positive effects into a new benchmark simulation.

The six identified areas can be summarised as follows:

(1) PDF-flamelet generation parameters
(2) Influence of different chemical mechanisms
(3) Different boundary conditions and the effect on wall heat transfer
(4) Influence of different turbulence models
(5) Turbulence parameters
(6) Transport modelling

These methods are summarised henceforth. The PDF-flamelet method and exploration into different chemical mechanisms are combined in Sec.6.2. Likewise, different turbulence parameters and models are detailed in Sec.6.5.

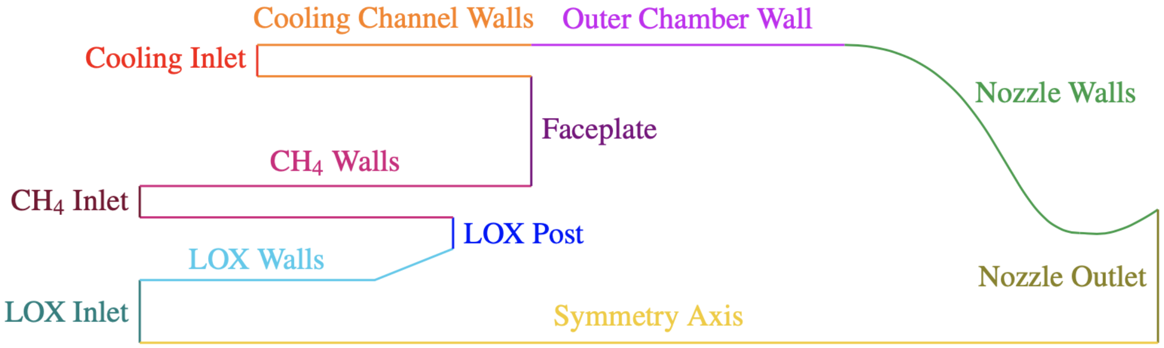


FIGURE 6.1. Labelled sections of BKN (not to scale), taken from [75]

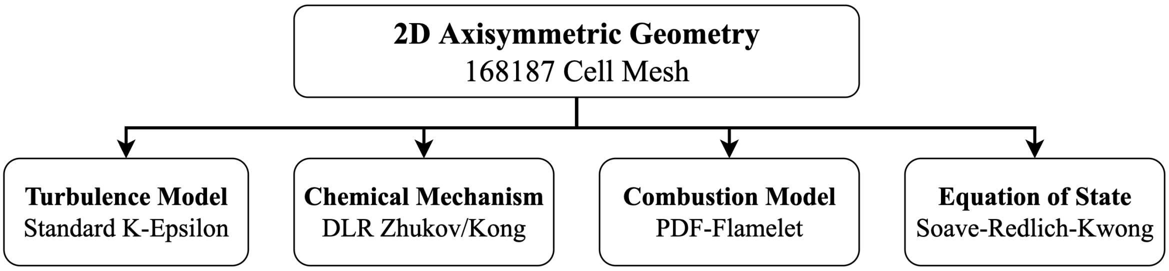


FIGURE 6.2. Core components of the BKN simulation completed by Morris

6.2 Combustion Modelling

6.2.1 Models Chosen

The PDF-flamelet approach was utilised as the turbulent combustion model in this work. Morris was able to use this method with the DLR Zhukov/Kong mechanism, however the parameters were left default. Thus, a priority was set on optimising these parameters in order to capture the chemistry as accurately as possible, namely the table resolution and stoichiometric scalar dissipation distribution. An evaluation of different mechanisms was also completed, with four schemes tested:

- DLR Zhukov/Kong - 23 species, 51 reactions
- RAMEC - 38 species, 190 reactions
- Reduced GRI3.0 - 35 species, 218 reactions
- DLR NewM - 26 species, 53 reactions

6.2.2 Flamelet Generation and Tabulation

As detailed in Chapter 5, the flamelet table generation relies on two key parameters, the table resolution and the stoichiometric scalar dissipation. The table resolution, often referred to as the number of gridpoints, discretises the mixture fraction into a monotonic range of values that are clustered around the stoichiometric mixture fraction. An increased resolution yields a more detailed representation of the temperature and species mass fractions at a given stoichiometric scalar dissipation value. The default number of gridpoints in Fluent is 32, which is compared to 128 in Fig.6.3. The stoichiometric mixture fraction value is approximately 0.2, indicated by the high concentration of mixture fraction values.

The distribution and number of stoichiometric scalar dissipation values is the other key consideration in the flamelet generation process. The lowest value should be selected as close to 0 as possible to represent chemical equilibrium, with range extending up until the point of flamelet extinction. Morris utilised a logarithmic distribution with 8 flamelet tables generated

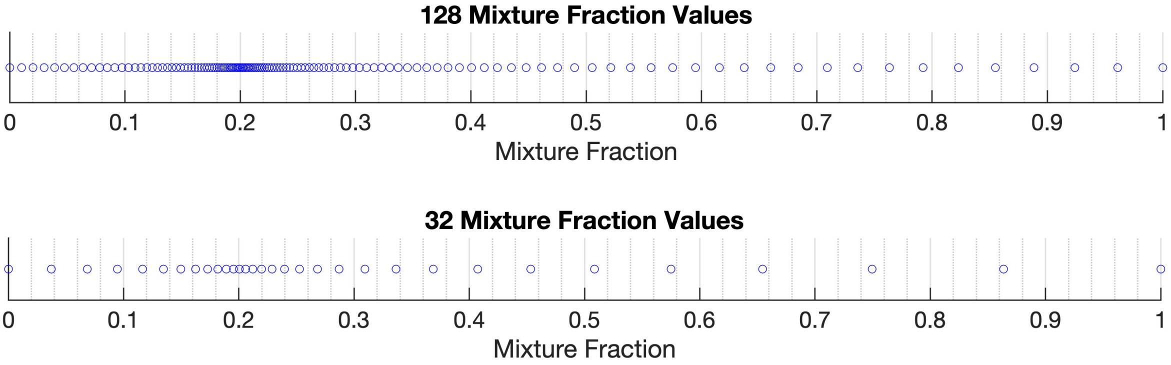


FIGURE 6.3. Concentrations of different table resolutions around the stoichiometric point

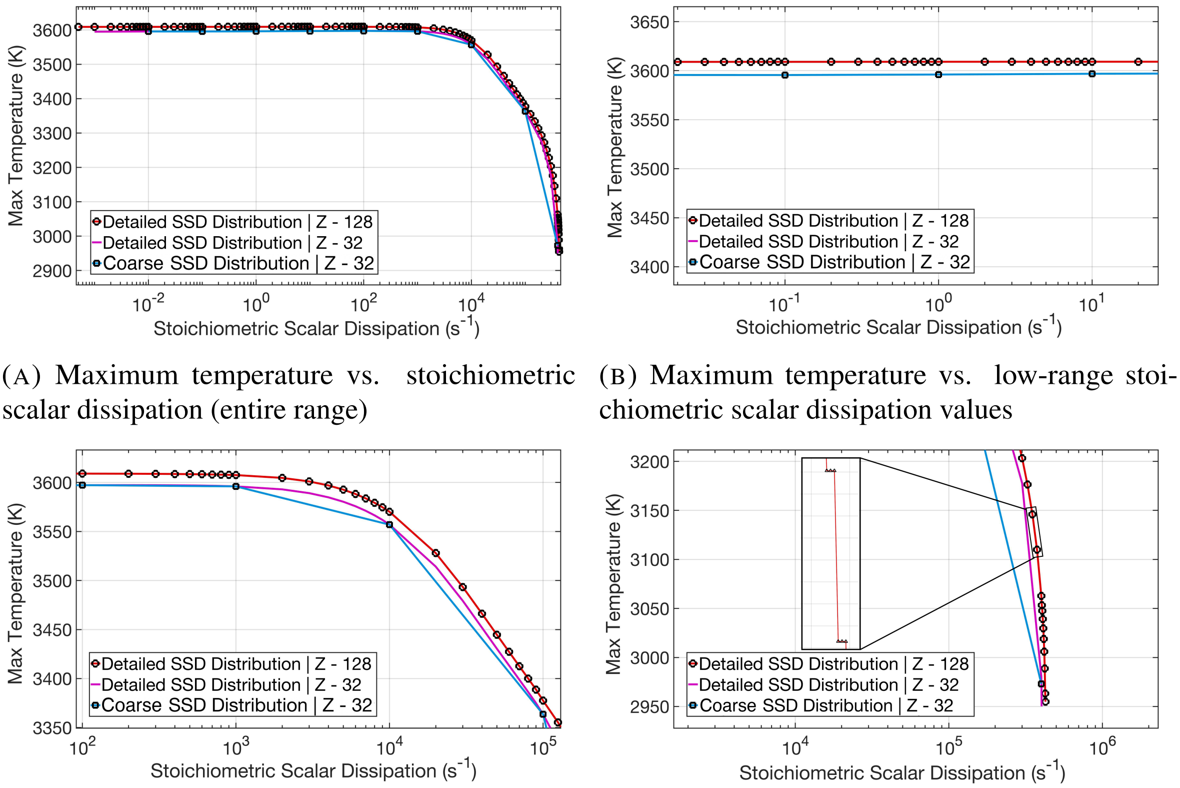
at stoichiometric scalar dissipation values of 0.01/s to 100,000/s with an additional table generated near the extinction point, at 400,000/s. Using 9 tables with a resolution of 32 as a benchmark, an investigation into the profiles of temperature and species mass fractions was conducted to properly evaluate the optimal parameters.

To begin, flamelet tables were generated in ANSYS Fluent using manual parameter control. The number of gridpoints, intial stoichiometric scalar dissipation ($\mathcal{X}_0$), the multiplier ($\lambda_{\mathcal{X}}$), the step ($\Delta\mathcal{X}$) and the maximum number of flamelets are all specified. Additionally, the equilibrium chamber pressure (66.8 bar) is specified for the density calculation in Eq.5.14 and Eq.5.15. Fluent will continue to generate tables until the maximum number is reached or flame extinction occurs, with the distribution dictated by Eq.6.1. The distributions achievable with this equation are limited, thus flamelets were generated individually or in pairs, and fused together in a text file later.

$$\mathcal{X}_i = \begin{cases} \lambda_{\mathcal{X}}\mathcal{X}_{i-1} & \text{for} \quad \mathcal{X}_{i-1} < 1/s \\ \mathcal{X}_{i-1} + \Delta\mathcal{X} & \text{for} \quad \mathcal{X}_{i-1} \geq 1/s \end{cases} \tag{6.1}$$

With a single flamelet text file, MATLAB was used to analyse the profiles of temperature and species mass fraction. A script was written that reads the flamelet file (Appendix C), sorts and arranges the data into parameter arrays and displays the desired profiles in 2D or 3D. 2D profiles were useful for plotting the maximum value of a certain parameter over the range of $\mathcal{X}_{st}$. 3D surface plots were used to capture the entire range of a parameter over both the mixture fraction and $\mathcal{X}_{st}$ ranges.

The change in maximum temperature over the range of scalar dissipation was a key focus, seen in Fig.6.4. The entire range is seen in Fig.6.4a, capturing a constant maximum temperature before the drop off closer to extinction. Fig.6.4b-6.4d highlights the individual areas of importance in the low, mid and high range $\mathcal{X}_{st}$ values. In Fig.6.4b, an increase to the table resolution from 32 to 128 yielded an increase in the maximum temperature. Additionally, increasing the resolution of $\mathcal{X}_{st}$ accurately captures the gradient shift in temperature, with the original logarithmic distribution profile coarse in comparison. Fig.6.4d focuses on the extinction point. A rise in the table resolution also increases the value of the $\mathcal{X}_{st}$ at which quenching occurs.



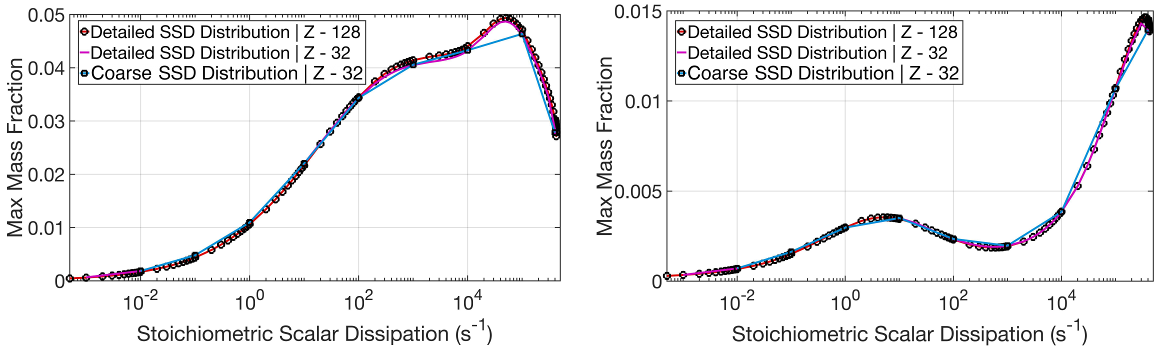
(A) Maximum temperature vs. stoichiometric scalar dissipation (entire range)

(B) Maximum temperature vs. low-range stoichiometric scalar dissipation values

(C) Maximum temperature vs. mid-range stoichiometric scalar dissipation values

(D) Maximum temperature vs. high-range stoichiometric scalar dissipation values

FIGURE 6.4. Maximum temperature vs. stoichiometric scalar dissipation



(A) C_2H_4 maximum mass fraction

(B) C_4H_6 maximum mass fraction

FIGURE 6.5. Maximum mass fraction vs stoichiometric scalar dissipation

As the flamelets approached extinction, smaller step values in the Fluent generator were used to find the exact value at which extinction occurs. This often caused redundant points to be generated, highlighted in Fig.6.4d. These were removed manually from the flamelet text file, once a final range was established.

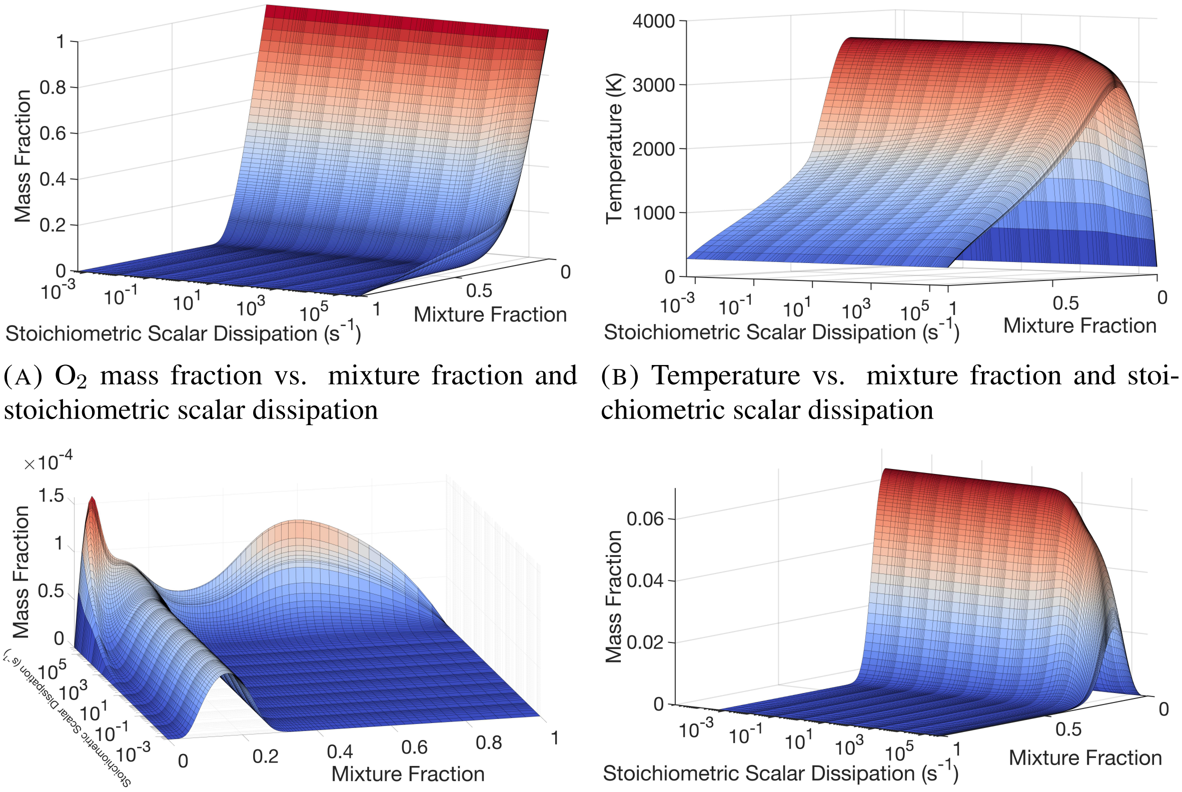
Species mass fraction profiles also constituted a key area of analysis. Maximum values were plotted in 2D, with many species exhibiting steep gradient shifts over the range of stoichiometric scalar dissipation. A more detailed distribution was required in order to capture the correct maxima and minima and avoid large value jumps, clear in Fig.6.5. An increase in the table resolution also imparted an upwards/downwards shift on the data points, a trend seen across most species. An increase beyond 128 yielded a negligible change in both temperature and mass fraction plots, hence it was chosen as the final table resolution. Its overall impact was minimal, with a shift of 0.4% on the maximum temperature.

As for the stoichiometric scalar dissipation distribution, the final range/distribution chosen can be summarised as follows:

- The starting point has been extended downwards from 0.01 to 0.0005.
- Each power of 10 (0.001-100000) contains a linear set of 10 values in the following pattern: ... 1, 2, 3... 10, 20, 30... 100, 200, 300...
- At 100000, a 25000 step is maintained until the flame is extinguished and the step needs to be reduced.

- The final value in the range is 425500, with 426000 corresponding to an extinguished flame. This is a 6.4% rise when compared to the benchmark value of 400000.

Within both Fig.6.4 and Fig.6.5 this distribution in conjunction with a table resolution of 128 is seen in red, with circular data points. Given the limits of the flamelet method, the chemistry is now captured as accurately as possible. All profiles exhibit smoother changes over a larger range, with correct maxima and minima. For further confirmation, 3D surface plots were used to capture the entire range of mass fractions and temperatures over both the mixture fraction and stoichiometric scalar dissipation. The newly captured detail of both parameters is observable with smooth profiles noted across all rapid fluctuations in temperature and mass fraction. Examples of temperature and the mass fraction of O_2, H_2O_2 and OH are provided in Fig.6.6. The optimised parameters were the basis for the DLR Zhukov/Kong mechanism and are therefore used for a comparison between different mechanisms.



(A) O_2 mass fraction vs. mixture fraction and stoichiometric scalar dissipation

(B) Temperature vs. mixture fraction and stoichiometric scalar dissipation

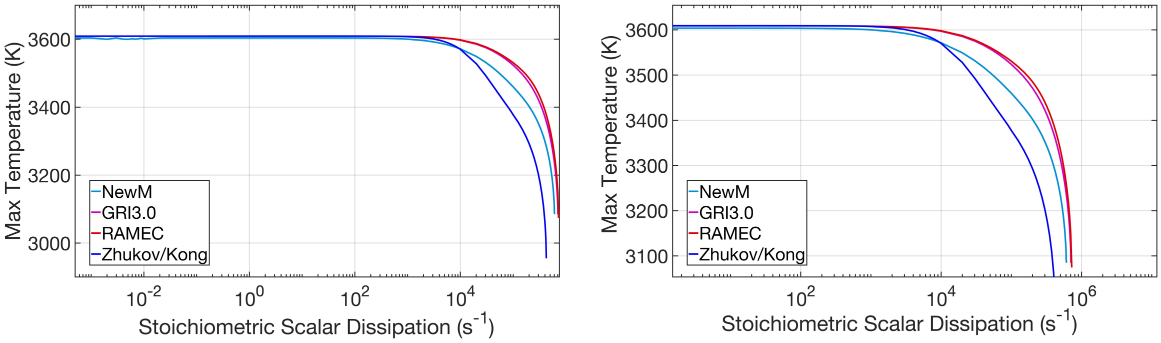
(C) H_2O_2 mass fraction vs. mixture fraction and stoichiometric scalar dissipation

(D) OH mass fraction vs. mixture fraction and stoichiometric scalar dissipation

FIGURE 6.6. Flamelet table 3D profiles of temperature and O2

6.2.3 Evaluation of Different Mechanisms

With an optimal flamelet table resolution and distribution defined, different mechanisms were compared. Fig.6.7 captures the maximum temperature profiles of the different mechanisms with a focus on the extinction point seen in Fig.6.7b. The constant maximum temperature in the lower range was almost identical across the different schemes, varying by a maximum of 8 K, however they all exuded different downwards trajectories in the upper range. The Zhukov/Kong mechanism exhibited a quicker drop off and a 3.9-4.2% lower max temperature in comparison to the other 3. The RAMEC mechanism did not extinguish until the stoichiometric scalar dissipation was above 725 000 1/s, a 70.4% rise when compared to the Zhukov/Kong scheme. The GRI3.0 was slightly lower at 710 000 1/s with the NewM at 610 000 1/s. In order to evaluate these schemes properly, simulation results are required to compare the flow field scalar dissipation range with the flamelet table range.



(A) Comparison of different mechanisms entire range of maximum temperature

(B) Comparison of different mechanisms extinction points

FIGURE 6.7. Comparison of different mechanisms maximum temperature vs. stoichiometric scalar dissipation

6.2.4 PDF Lookup Table Generation

With the chemistry captured in the flamelet tables, the PDF integration and table generation process is undertaken to form a complete lookup table for mean temperature, density and species mass fraction. The Fluent automated grid refinement algorithm was used, which creates a coarse grid using a specified number of initial grid points. It then inserts additional points if the change in value or slope breaches a defined threshold. Eq.6.2 defines the change in value threshold, with a point inserted between $\mathcal{V}_i$ and $\mathcal{V}_{i+1}$ if:

$$|\mathcal{V}_i - \mathcal{V}_{i+1}| > \epsilon_{\mathcal{V}}(\mathcal{V}_{max} - \mathcal{V}_{min}) \tag{6.2}$$

where ϵ_V is the maximum change in value ratio and $\mathcal{V}_{max}$ and $\mathcal{V}_{min}$ are the maximum and minimum values over the whole grid range respectively. The same process is applied to the slope, where ϕ is the independent grid variable (mean mixture fraction, mean mixture fraction variance or mean enthalpy).

$$|\mathcal{S}_i - \mathcal{S}_{i+1}| > \epsilon_{\mathcal{S}}(\mathcal{S}_{max} - \mathcal{S}_{min}) \quad \text{where} \quad \mathcal{S}_i = \frac{\mathcal{V}_{i+1} - \mathcal{V}_i}{\phi_{i+1} - \phi_i} \tag{6.3}$$

These processes ensure gradual changes between successive gridpoints. Each PDF table is parameterised by a mean scalar dissipation value, similar to the flamelet tables. The grid refinement parameters used are shown in Fig.6.1.

TABLE 6.1. PDF table automated grid refinement parameters

Parameter	**Input**
Initial Number of Gridpoints	32
Maximum Number of Gridpoints	200
Maximum Change in Value Ratio	0.25
Maximum Change in Slope Ratio	0.25

6.3 Wall Heat Transfer

6.3.1 Combustion Chamber Wall Boundary Treatment

Fluent offers numerous ways to define the wall boundary conditions. In the previous work, Morris left with the chamber walls as adiabatic. This enforces no heat transfer. To allow of wall heat flux, temperature profiles were applied based on experimental data. By prescribing these profiles at the wall, Fluent can calculate a resultant heat transfer. Two sets of temperature measurements were recorded during the BKN test fire, with the thermocouples positioned as follows [67]:

- 0.1 mm into the flow, situated every 20 mm from 54.5 mm to 294.5 mm
- 1 mm behind the fluid-side surface, situated every 20 mm from 34.5 mm to 304.5 mm

Using the data sets, two different profiles were developed. MATLAB was used to fit polynomials to both sets of data using the `polyfit` function. For the 0.1 mm measurements an 8th order polynomial was selected as it traced the multiple gradient shifts accurately. A piece-wise function was used to attach constant segments on either end to cover the entire chamber wall. Similarly, a piecewise function was used to combine a constant function, a 3rd order polynomial and linear function to create a temperature profile for the embedded sensors. Both profiles and the experimental datapoints can be seen in Fig.6.8.

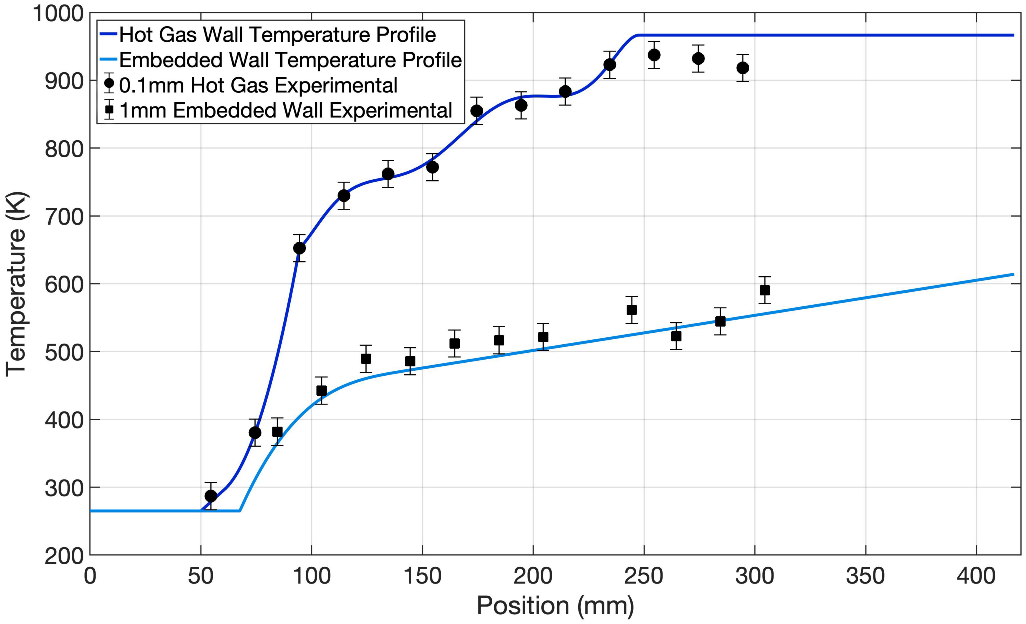


FIGURE 6.8. Fitted wall temperature profiles to experimental values

Here, an error bar of 20 K is applied to the experimental datapoints. Note, the 8th embedded thermocouple at x = 224.5 mm was faulty thus its reading was removed from the data set. The constants applied in the first 50 mm were set to 265 K, the temperature of the injected film coolant. The coolant injector is 9 mm in length thus Fluent doesn't use values from 0-41 mm. The faceplate is located at x = 50 mm, the point from which the chamber wall begins, see Fig.6.1. The nozzle wall starts at 372 mm and as no measurements were taken in this region, it wasn't considered for heat transfer analysis. The 0.1 mm profile extends a constant value from the peak of the 8th order polynomial, as it was expected that the temperature would not decrease, suggested by the final 3 measurements. The embedded measurements didn't exude major gradient shifts from 150 mm, thus a linear segment was added from the 4th measurement onwards and left to continue through the nozzle. No nozzle temperatures or heat flux measurements were recorded, thus it was just assumed the wall temperature would continue to increase.

6.3.2 Temperature Profile UDF's

Once the profiles were generated in MATLAB, a method of importing them into Fluent was required. C code user-defined-functions (UDF's) are utilised by Fluent to allow the incorporation of custom wall profiles written externally. More detail on C UDF's is provided in Sec.6.4, with the temperature profile functions located in the Appendix D.

6.3.3 Boundary Condition Setup

The hot gas profile was imported and fixed as the wall temperature, with the wall modelled as smooth with no slip. No wall material properties or thickness were required. Fluent calculates the heat flux using the applied profile as follows:

$$q = \kappa \frac{\partial T}{\partial n}(T_w - T_f) \tag{6.4}$$

Here n is the local coordinate normal to the wall, T_w is the wall temperature specified by the applied profile and T_f is temperature of the fluid in the adjacent boundary cells. Fluent uses the law-of-the-wall to compute the fluid temperature, detailed in the Fluent theory guide [49].

The embedded profile allows for a different approach to modelling the wall heat flux. Unlike the hot gas profile, the temperature is set behind a 1 mm nickel-alloy inconel 600 wall from which Fluent uses a simple 1D conduction formula to calculate the thermal resistance (Eq.6.5). Here, a constant wall thermal conductivity (κ_w) of 22 W/mK was used [2]. Instead of fixing a temperature boundary, a gradient is applied and Fluent calculates the wall heat flux and wall temperature simultaneously.

$$\Omega_w = \frac{\Delta x}{\kappa_w} \tag{6.5}$$

6.4 Transport Modelling

As discussed in Sec.4.5.2, the transport properties of viscosity and thermal conductivity have been modelled using the CEA formulation. Each species has its own temperature dependant equation built using four coefficients. The viscosity and thermal conductivity of the entire fluid are constructed using mixing laws. These equations are repeated here for clarity.

$$\left.\begin{matrix} ln\mu \\ \\ ln\kappa \end{matrix}\right\} = AlnT + \frac{B}{T} + \frac{C}{T^2} + D \tag{4.52 revisited}$$

$$\mu_m = \sum_K \frac{V_K \mu_K}{\sum_k V_K \phi_{K,k}} \quad \text{where} \quad \phi_{K,k} = \frac{1}{\sqrt{8}} \left(1 + \frac{Y_K}{Y_k}\right)^{-0.5} \left[1 + \left(\frac{\mu_K}{\mu_k}\right)^{0.5} \left(\frac{Y_k}{Y_K}\right)^{0.25}\right]^2 \tag{4.53 revisited}$$

$$\kappa_m = \sum_K \frac{V_K \kappa_K}{\phi_K} \quad \text{where} \quad \phi_K = \sum_K V_k \sqrt{\frac{Y_K}{Y_k}} \tag{4.54 revisited}$$

The individual species functions and the two mixing laws are defined using C code UDF's that are imported into Fluent. As each mechanism contains at least 23 species, an automated approach to generating each UDF file was developed. This was executed using a MATLAB script (Appendix C) that combined all the necessary processes together. The procedure can be divided into the following steps:

(1) The entire CEA database text file is imported into MATLAB.

(2) The chosen mechanism is imported into MATLAB.

(3) The mechanism species are identified.

(4) The first species is located in the CEA file and its coefficients are extracted.

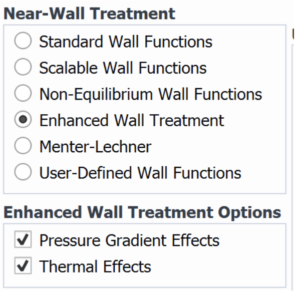
(5) Using the coefficients, a UDF is written for both viscosity and thermal conductivity in C.

(6) Steps 4 and 5 are repeated for all species in the mechanism.

A breakdown of the algorithm is given in Appendix A1. Additionally an overview of ANSYS Fluent UDF's utilising the viscosity mixing law as an example is given in Appendix A2.

6.5 Turbulence Modelling

6.5.1 Boundary Flow



Near-Wall Treatment

○ Standard Wall Functions
○ Scalable Wall Functions
○ Non-Equilibrium Wall Functions
● Enhanced Wall Treatment
○ Menter-Lechner
○ User-Defined Wall Functions

Enhanced Wall Treatment Options

☑ Pressure Gradient Effects
☑ Thermal Effects

FIGURE 6.9. Boundary flow options within ANSYS Fluent

Standard wall functions were utilised by Morris, however both the theory manual and existing works highlight the importance of using either the EWT or Menter Lechner formulations, detailed in Chapter 4. Due to the limited evidence to support the superiority of either option, both were compared using otherwise identical simulation setups. Fluent offers a checkbox within the turbulence model menu for interchanging between different types of near-wall treatments. They are only available within the k-ϵ class of turbulence models, with all options listed in Fig.6.9. When using the EWT, the pressure gradient and thermal effects were both turned on. The influence of different near-wall treatments is presented in Sec.7.1.

6.5.2 Turbulent Schmidt Number

Within the same menu window, the PDF turbulent Schmidt number can be modified. As previously mentioned in Sec.4.4.6, a comparison between three values, 0.85 (default), 0.7 and 0.55 was made. In Chapter 3 it was noted that the turbulent Schmidt number had a significant effect on wall values, thus a simulation with accurate boundary flow and heat transfer modelling was established first. From here, three identical simulations were run using the aforementioned turbulent Schmidt numbers. A detailed description of the simulation progression is presented in Sec.6.6. The results are seen in Sec.7.1.2.

6.5.3 Alternative Turbulence Models

The standard k-ϵ model has been chosen as the primary turbulence model due it to its abundance/accuracy in literature and successful implementation in the Morris simulation. Two additional models were used for comparison purposes; the RNG k-ϵ model and the GEKO model. The RNG model was setup with default coefficients and the differential viscosity option turned on, described in Sec.4.4.2. The GEKO model uses a set of free parameters to emulate different turbulence models, with a total of seven different parameter combinations outlined in Tab.6.2.

TABLE 6.2. GEKO turbulence model coefficients

Simulation No.	CSEP	CNW	CMIX	CJET	CBF_TUR
1	1	1	0	0	2
2	1	1	0.75	0.9	2
3	1	1	0.75	0.9	1.5
4	0.75	1	0.75	0.9	1.5
5	0.75	1	1	0.9	1.5
6	0.75	1	1	0.9	1.25
7	0.75	1	1	0.9	1

The first simulation uses standard k-ϵ equivalent parameters. Strokach *et al.* noted an improved conformity to experimental data through the increase of the CMIX parameter, hence simulation 2 increases CMIX to 0.75. CJET is a submodel of CMIX, therefore it only becomes a consideration when CMIX is non-zero. The Fluent user guide also provides simulation results, with a default CJET value of 0.9 matching experimental data the closest [50]. Simulations 3-7 were performed purely to test the limits of the GEKO model and the effect on the flame shape. CBF_TUR is an auxiliary parameter that on reduction increases the effect of CMIX and CJET near the wall. Through initial experimentation it was observed that this parameter had a large impact on the overall flame shape, thus its effect was formally documented. For a detailed description on the role of each parameter, refer to the theory guide [49]. The results for simulation 1 are presented in Sec.7.4. The remaining simulation results are presented in Appendix B.

6.6 Simulation Setup

6.6.1 Simulation Roadmap

Throughout the course of this thesis, numerous simulations were run. A simulation roadmap was established, so that a logical progression through the defined areas of exploration could be executed. The effects of each change are noted, allowing a sequential analysis of the implemented changes. Additionally, each simulation is saved, allowing backtracking at any point. This is useful when older simulations with simpler setups and faster calculation times were required to check the difficult modelling approaches. For example, different mechanisms were tested later in the project. Backtracking to a model prior to the inclusion of complex transport modelling helped in keeping the computation time reasonable.

The six areas of exploration and their simulations are detailed in the Fig.6.10. Colour coding was used to track progress, with the example shown below taken from the latter stages of the project. The combined positive effects from each category were carried forward into a final simulation, satisfying the aim of producing a new simulation of high fidelity. Whilst most facets of the simulation were manipulated, the equation of state remained constant. The SRK EoS was chosen, informed by both its use in literature and the successful use in the Morris simulation. Additionally, all simulations utilised the mesh developed by Morris [75].

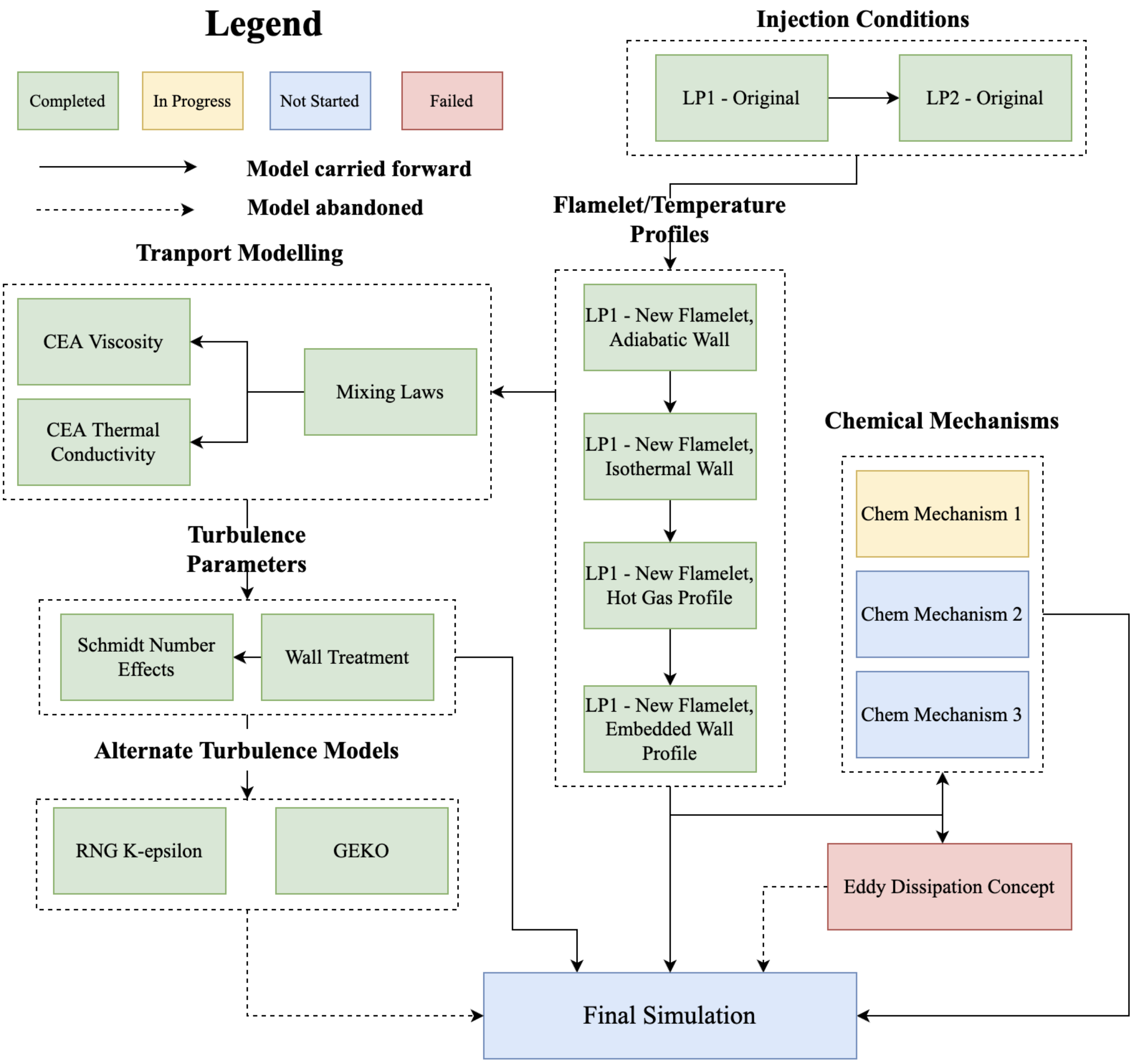


FIGURE 6.10. Simulation roadmap

6.6.2 Simulation Convergence

A consistent procedure was followed to aid in solution convergence. When the actual conditions of BKN are incorporated from the first iteration, the solution quickly diverges yielding floating point errors. To combat this, the following steps were followed:

(1) Setup a first order calculation with all the correct models and boundary conditions. Do not import any UDF's at this point.

(2) Change the LOX injection temperature from 121K (load point 1) to 250K and the mixture density method from SRK to PDF.

(3) Initialise the solution to "all zones" and start the calculation

(4) Once the residuals have flattened out (approximately 500-1000 iterations), change the LOX temperature back to 121K

(5) When the residuals drop back down (approximately 250-500 iterations), change the material density back to SRK

(6) When the residuals drop back down (approximately 500-1000 iterations), import the UDF's and apply them to the simulation

(7) Finally, change the solution method to second order and let the simulation run to convergence.

The steps listed here were adhered to as a safe approach for the more complex simulations. Simpler UDF's such as the temperature profiles, usually caused no problems when incorporated from the start. The SRK step was the most important, with divergence occurring every time it weren't followed.

Under-relaxation factors also played a large role in stabilising the solution. Divergence errors were eliminated when reducing these values (often as low as 0.1), albeit with slower convergence. The threshold for convergence was set to 10^{-5} for all residuals. Oscillatory behaviour was consistent once the solution had converged, usually after 3000-4000 iterations.

6.6.3 Simulation Guide

Tab.6.3 is provided as a reference for all simulation results presented in Sec.7. Whilst the figure legend entries are indicative of the sequential changes made, for clarity the entire simulation setup is provided here. Some results double up and in the context of the figure have different legend entries, indicated by brackets under simulation name. For clarity, snippets of this table are included throughout Chapter 7.

TABLE 6.3. Complete set of simulations

Simulation Name	**LP**	**EoS**	**Turb. Model**	**Combustion Model**	**Chemical Mechanism**	**Transport Mixing**	**Transport Species**	**Wall Profile**	**Boundary Flow**	**Sct**
LP1 - Original - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP2 - Original - Adiabatic	2	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP1 - Detailed Flamelet - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP1 - MW/EW - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Adiabatic	Enhanced Wall Treatment	0.85
LP1 - Hot Gas Profile	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Constant	Hot Gas	Standard Wall Functions	0.85
LP1 - Hot Gas Profile - MW	1	SRK	k-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Standard Wall Functions	0.85
LP1 - Hot Gas Profile - MW/EW	1	SRK	k-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Enhanced Wall Treatment	0.85
Menter Lechner	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Menter Lechner	0.85
LP1 - Embedded	1	SRK	k-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.85
SCT - 0.70 - MW	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.70
SCT - 0.55 - MW	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.55
SCT - 0.55 - Wilke/Zipp	1	SRK	k-ϵ	PDF-Flamelet New Profile	DLR Z/K	Wilke/Zipp	CEA	Embedded	Enhanced Wall Treatment	0.55
GEKO	1	SRK	GEKO	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Adiabatic	Standard Wall Functions	0.85
RNG	1	SRK	RNG	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Adiabatic	Standard Wall Functions	0.85
DLR Z/K	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
NewM	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	NewM	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
GRI3.0	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	GRI3.0	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
RAMEC	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	RAMEC	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
Final	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	NewM	Wilke/Zipp	CEA	Embedded	Enhanced Wall Treatment	0.70

6.6.4 Injection Conditions

The utilised injection conditions replicate the two load points used in the experimental reference for this work [67]. A simulation was completed using the original configuration (same as the final Morris simulation) for each load point, with load point 1 carried forward for the remainder of the thesis. The temperature of the film cooling inlet has been reduced due the geometry simplification, justified by Morris [75].

TABLE 6.4. Experimental load point injection conditions

Parameter	**Units**	**LP1**	**LP2**
$\dot{m}_o$	kg/s	0.362	0.365
$\dot{m}_f$	kg/s	0.129	0.123
$\dot{m}_c$	kg/s	0.298	0.290
T_o	K	121	125
T_f	K	279	277
T_c	K	265	262
P	bar	66.8	66.2
ROF		2.8	3.0
VR		17.1	15.6
J		15.8	13.6

CHAPTER 7

Results

This chapter presents the results from all major simulations outlined in Tab.6.3 in comparison to experimental data. Sec.7.1 presents the simulation results obtained along the chamber walls, including heat transfer, temperature and pressure. Sec.7.2 exhibits the axial results, indicative of the overall flame length. Sec.7.3 presents the effect of different chemical mechanisms with Sec.7.4 highlighting flow field differences between turbulence models. Finally Sec.7.5 collates the results from the final simulation and compares them to the original setup.

7.1 Wall Results

7.1.1 Boundary Condition Variation

Fig.7.1 presents the total temperature calculated at the wall for all simulations that used adiabatic walls in comparison to the developed temperature profiles and experimental measurements. All adiabatic results are clustered in the middle, with Fig.7.1b highlighting the differences between them. The two experimental load points yield a 10 K difference (orange and yellow), with the detailed flamelet dropping the temperature by 20 K (blue). When the enhanced wall treatment and CEA modelling is incorporated with mass weighted mixing laws, the temperature jumps by 40 K (green).

The differences between the results are negligible when viewed in context of the applied temperature profiles. The hot gas profile is 200-300 K higher than the highest adiabatic result. Heat transfer is modelled when the temperature profile is applied, thus the adiabatic profiles should be hotter as heat escape is prohibited. A possible explanation of this result is that the

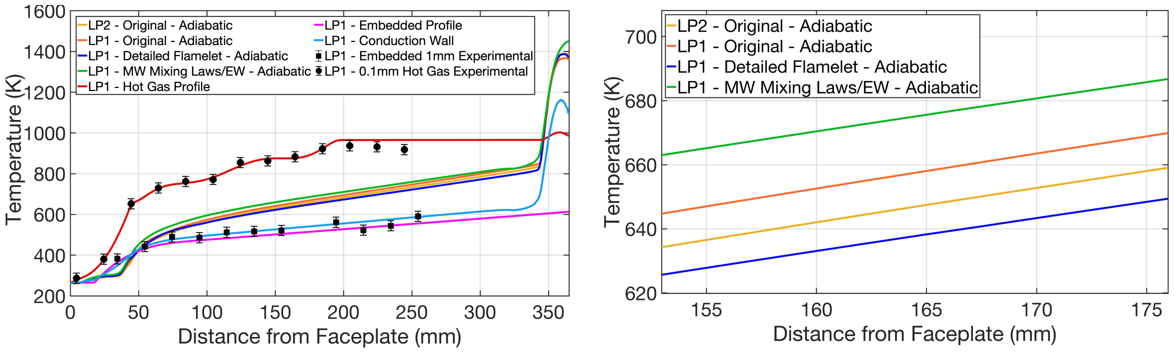
TABLE 7.1. Variable boundary conditions simulation setup

Simulation Name	LP	EoS	Turb. Model	Combustion Model	Chemical Mechanism	Transport Mixing	Transport Species	Wall Profile	Boundary Flow	Sct
LP1 - Original - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP2 - Original - Adiabatic	2	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP1 - Detailed Flamelet - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
LP1 - MW/EW - Adiabatic	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Adiabatic	Enhanced Wall Treatment	0.85
LP1 - Hot Gas Profile	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Constant	Hot Gas	Standard Wall Functions	0.85
LP1 - Hot Gas Profile - MW (Wall Functions)	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Standard Wall Functions	0.85
LP1 - Hot Gas Profile - MW/EW (Enhanced Wall)	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Enhanced Wall Treatment	0.85
Menter Lechner	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Hot Gas	Menter Lechner	0.85
LP1 - Embedded (SCT - 0.85 - MW)	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.85

slight protrusion of the thermocouples has impeded the boundary flow, causing collisions that subsequently raise the experimental temperature measurements higher than an unimpeded flow.

This result is fixed when the embedded temperature profile is applied and a conduction gradient is used to calculate wall temperatures and heat fluxes. From 50 mm onwards, the result is lower than adiabatic, as expected. Both sets of experimental measurements rise in temperature in the fore of the chamber quicker than the numerical results. This implies that perhaps the chamber wall before ignition was warmer than ambient temperature. The film cooling at 265 K (load point 1) has a higher impact in decreasing the wall temperature rise within the simulations.

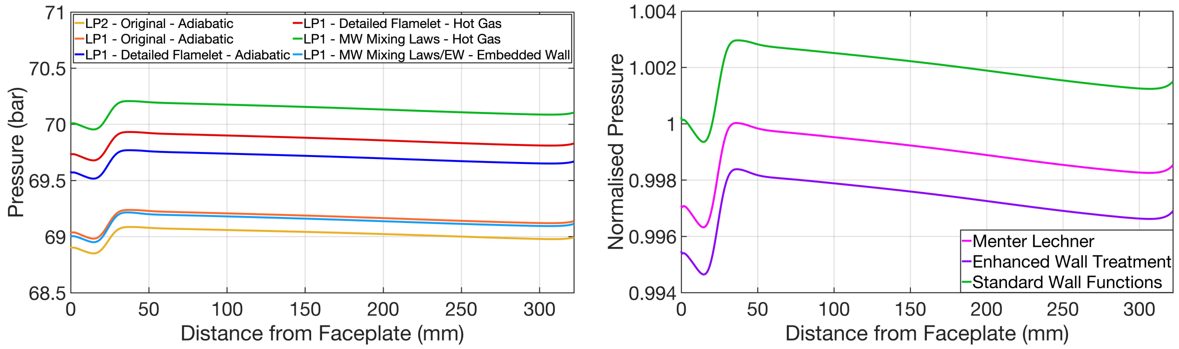
Looking at the wall pressures in Fig.7.2a, the optimised flamelet tables induce a 0.5 bar increase when compared to the original setup (blue). When the hot gas wall temperature profile is applied, the pressure increases further, consistent with the large increase in wall temperature (red/green). A drop from 70 bar to 69 bar is observed when the embedded temperature profile is used with enhanced wall treatment, attributed to the more accurate



(A) Comparison of wall temperatures with experimental datapoints

(B) Adiabatic wall temperature differences

FIGURE 7.1. Wall temperature results from all simulations with adiabatic walls and initial simulations with temperature profiles



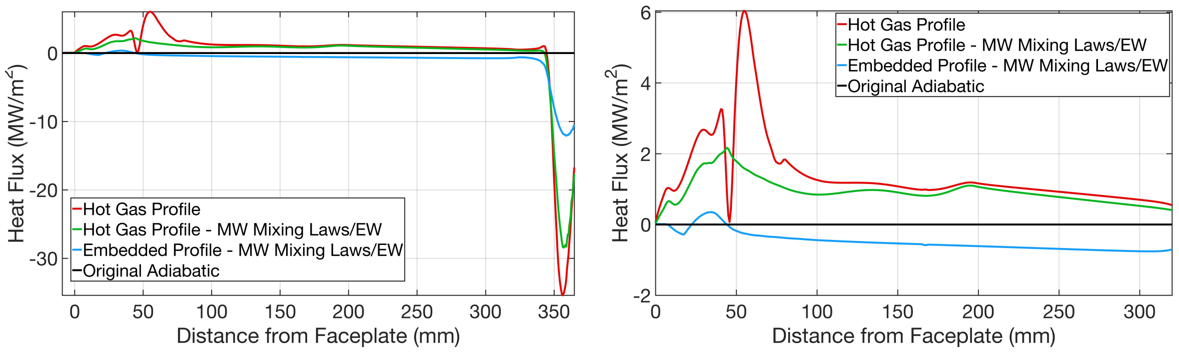
(A) Pressure comparison between adiabatic walls and temperature profiles

(B) Menter Lechner vs EWT wall pressure

FIGURE 7.2. Initial wall pressure results

temperatures at the wall, seen in Fig.7.1a (light blue). Additionally, this result moves closer to experimental values of approximately 66.6 bar (see Fig.7.4b). An evaluation of the Menter Lenchner and enhanced wall treatment formulations is provided in Fig.7.2b, normalised to highlight the 0.2% difference. The equivalent simulation using standard wall functions is also provided (green). As seen in literature, the difference between the two methods is minor. Enhanced wall treatment shows the best conformation to experimental pressures, justifying its choice in this work.

The wall heat flux results are presented in Fig.7.3 and are intertwined with pressure and temperature. As seen in Fig.7.3b, the flux for the hot gas profiles over the outer wall segment (0-322mm) is positive (red/green), suggesting that heat is being delivered to the chamber. This



(A) Nozzle/outer wall heat flux comparison

(B) Outer wall heat flux comparison

FIGURE 7.3. Temperature profile heat flux comparison

confirms the invalidity of the wall temperature results produced using the hot gas temperature profile. Conversely, the embedded wall temperature profile shows an expected net heat loss over the chamber wall. There is a small positive segment in the first 50mm due to a quicker rise in experimental temperatures. This crossover of the adiabatic line corresponds to the point where the embedded wall temperature profile exceeds the calculated wall temperature, discernible at the first experimental data point in Fig.7.1a. All profiles exhibit a negative heat flux out the nozzle.

7.1.2 Turbulent Schmidt Number Variation

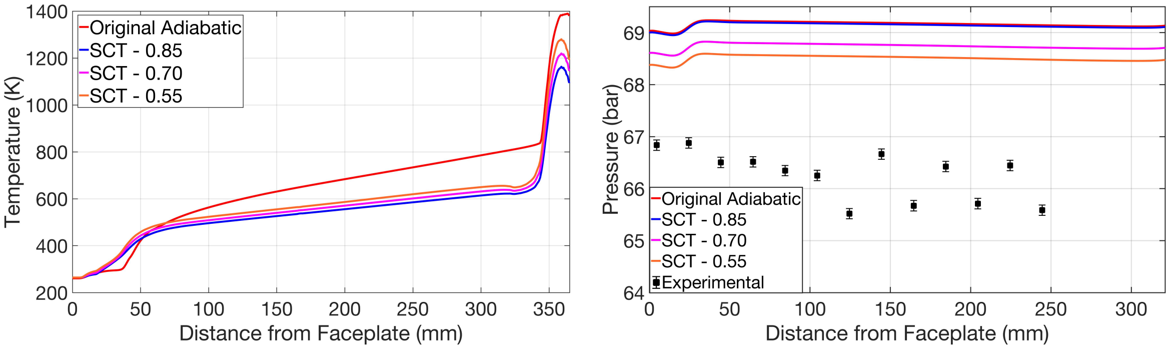
TABLE 7.2. Simulation setup for turbulent Schmidt number comparison

Simulation Name	LP	EoS	Turb. Model	Combustion Model	Chemical Mechanism	Transport Mixing	Transport Species	Wall Profile	Boundary Flow	Sct
LP1 - Embedded (SCT - 0.85 - MW)	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.85
SCT - 0.70 - MW	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.70
SCT - 0.55 - MW	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.55

As mentioned in Sec.6.5.2, the turbulent Schmidt number was manipulated once a more detailed simulation was setup, detailed in Tab.7.2. As the PDF-flamelet model assumes the Lewis number to be equal to 1, the turbulent Prandtl number is enforced to equal the turbulent Schmidt number. Lowering the turbulent Schmidt number (and the turbulent Prandtl number) was expected to raise the wall temperatures, as it enhances species and thermal diffusion and thus mixing and combustion. This is confirmed in Fig.7.4a with a 15 K rise observed for each reduction of 0.15. These results are in agreement with the findings in literature, noting intensified combustion and higher temperatures.

When viewing the effects of smaller turbulent Schmidt numbers on wall pressure, seen in Fig.7.4b, an opposite trend to that of temperature is noted, with pressures dropping further towards experimental. No difference in the gradient was observed, unlike the results presented by Chemnitz [84]. The dropping wall pressures are coupled with a rise in the net heat flux out of the chamber, seen in Fig.7.5. The integral of the wall heat fluxes yields the total heat transfer, captured in Tab.7.3. As the turbulent Schmidt number is lowered, the results increase towards the experimental value of -78 kW, a trend also noted in literature [84, 15, 52, 46]. Even with a turbulent Schmidt number of 0.55, the total heat transfer deviates from experimental by 41.8%.

Whilst the positive area in Fig.7.5 has been reduced, the gradient shift and concavity change between 20 and 50mm still inflicts a large reduction on the total heat transfer out of chamber. The discrepancy can also be attributed to simplifications in the simulation. The conduction gradient is only 1D, limiting the total heat transfer. There is also a lack of any recombination reactions in the boundary layer, a limitation of the flamelet model. Additionally, other works have extended the non-adiabatic flamelet model to include a permeable thermal boundary condition within mixture fraction space, significantly improving wall heat flux results [61].



(A) Different turbulent Schmidt number's effect on wall temperature

(B) Different turbulent Schmidt number's effect on wall pressure

FIGURE 7.4. Effects of lower turbulent Schmidt numbers on wall results

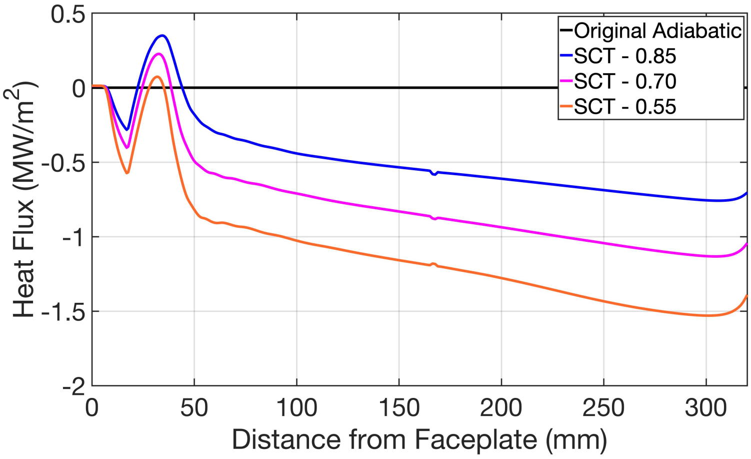


FIGURE 7.5. Different turbulent Schmidt number's effect on wall heat flux

TABLE 7.3. Total heat transfer results

Simulation	**Total Heat Transfer (kW)**
Sct 0.85 - MW	-24
Sct 0.70 - MW	-38
Sct 0.55 - MW	-55
Experimental	-78

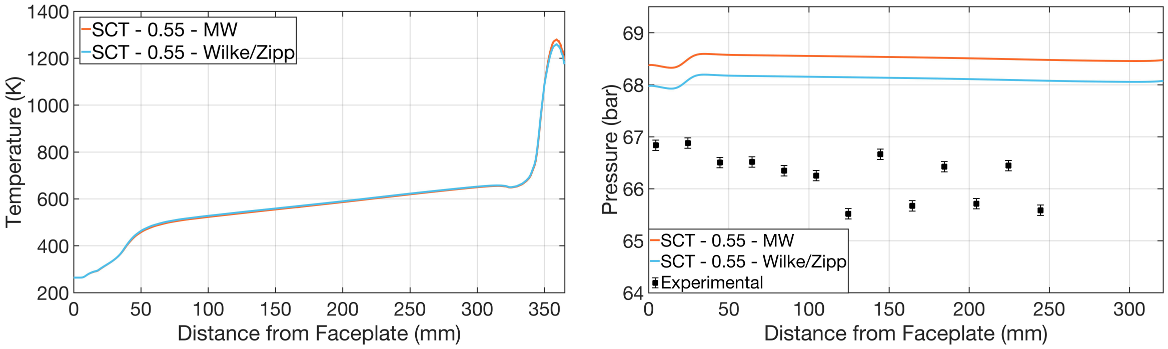
7.1.3 Transport Mixing Law Comparison

TABLE 7.4. Simulation setup for mixing law comparison

Simulation Name	**LP**	**EoS**	**Turb. Model**	**Combustion Model**	**Chemical Mechanism**	**Transport Mixing**	**Transport Species**	**Wall Profile**	**Boundary Flow**	**Sct**
SCT - 0.55 - MW	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.55
SCT - 0.55 - Wilke/Zipp	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Wilke/Zipp	CEA	Embedded	Enhanced Wall Treatment	0.55

The Wilke and Herning/Zipperer mixing laws for viscosity and thermal conductivity (Eq.4.53 and Eq.4.54) have been compared with mass weighted mixing laws. The simulation setup is defined in Tab.7.4. In Fig.7.6a a negligible difference between wall temperatures is observed. Fig.7.6b highlights a 0.4 bar difference in wall pressure, reducing the deviation with experimental to 1.8%. Mass weighted laws may be used to save computational cost, however they diminish the effects of the lighter species such as hydrogen. H_2 was found to be a dominant species in the boundary flow, hence the more complex mixing laws incorporate its influence on the mixture more realistically, improving pressure results.

The drop in pressure has yielded an increase in wall heat flux, matching the trends seen with the turbulent Schmidt number variation. Unlike in Fig.7.5, a key difference is seen in the first 45 mm of Fig.7.7. The trajectories remain the same as the injected CH_4 is the only species present in the boundary flow. At approximately 45 mm downstream, additional species enter the flow, with the film coolant reducing in dominance. The Wilke/Zipp mixing laws capture the mixture more realistically, with the flux tending further towards experimental. The total heat transfer is -59 kW, a 24.3% discrepancy with experimental.



(A) Mass Weighted vs. Wilke and Zipperer/Herning mixing laws effect on wall temperature

(B) Mass Weighted vs. Wilke and Zipperer/Herning mixing laws effect on wall pressure

FIGURE 7.6. Effects of different mixing laws on wall results

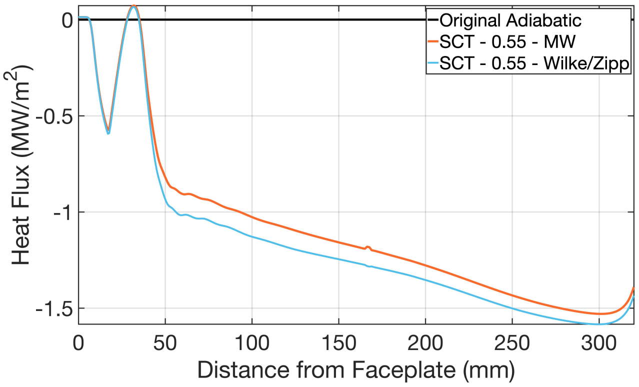
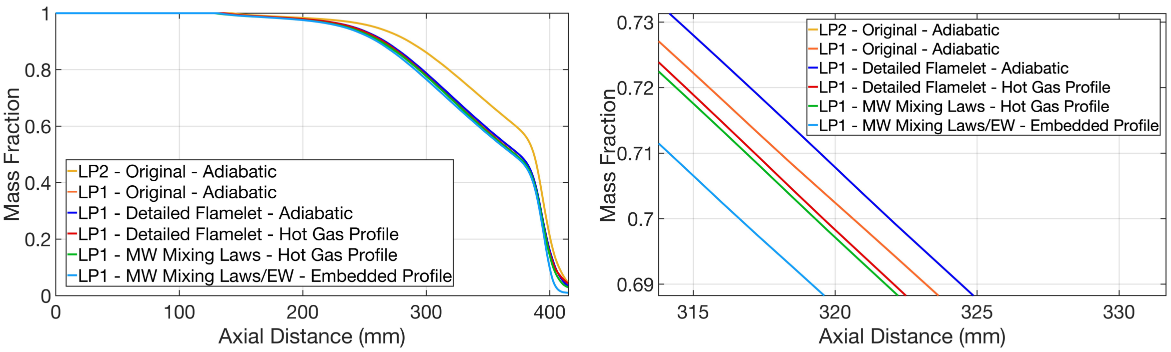


FIGURE 7.7. Mass Weighted vs. Wilke and Zipperer/Herning mixing laws effect on wall heat flux

7.2 Axial Results

Results recorded along the axis of symmetry have been analysed as indicators of the flame length and shape. The mass fraction of oxygen along the axis tracks how much has been burnt, indicative of mixing and combustion efficiency. In Fig.7.8a it can be seen that injection conditions influence the result the most. The higher momentum flux and velocity ratio in LP1 has increased mixing and consumption, a relationship also observed in both numerical and experimental works, discussed in Sec.3.4.1.3. When examining the individual effects of added detail to LP1 simulations (see Tab.7.1), minor differences are seen. All profiles exude similar gradients, separated by a maximum of 5 mm (see Fig.7.8b). Similar trends are observed in the axial temperature results. LP1 simulations have a 130-210 K higher peak when compared to LP2, with hotter peaks moving further upstream with each sequential change.

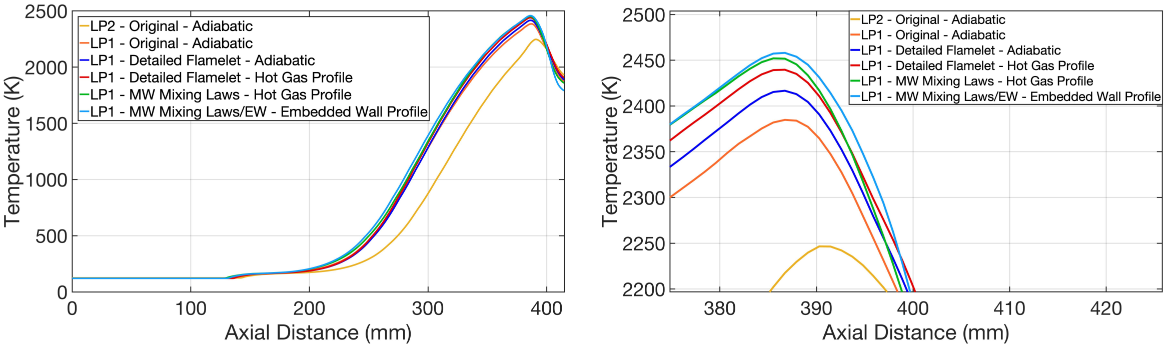
When incorporating the detailed flamelets (dark blue), the higher maximum temperature in the flamelet tables (see Fig.6.4) also causes a rise in axial peak temperature. As the hot gas profile results have been discounted as unrealistic, they will be ignored. Interestingly, the addition of the embedded profile induces the highest peak temperature (light blue). In contrast, at the wall the temperature drops significantly, attributed to the heat flux out of the chamber. Higher fluxes are the consequence of enhanced mixing and shorter flames, thus an increase in temperature with a peak further upstream is expected. This is relationship is also seen on reduction of the turbulent Schmidt number, which was previously linked to higher wall heat fluxes. In Fig.7.10a, the axial temperature has increased alongside a peak further upstream.



(A) Axial O2 mass fraction (entire range)

(B) Axial O2 mass fraction (315-330mm)

FIGURE 7.8. Axial O2 mass fraction

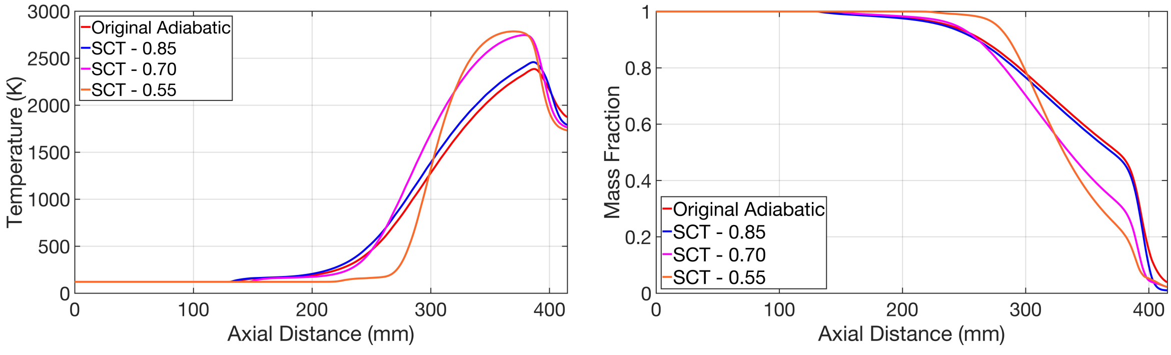


(A) Axial total temperature (entire range)

(B) Axial total temperature peaks

FIGURE 7.9. Axial total temperature

Coupling between the point at which the oxygen starts burning and temperature begins to rise is evident in Fig.7.10b. A turbulent Schmidt number of 0.55 displaces this point 90mm further downstream when compared to 0.85, however the rate at which oxygen is burnt is far higher. Overall, this result implies that flame reduces in length with a lower turbulent Schmidt number, a result consistent with literature [84]. Due to the unitary Lewis number assumption, the indirect lowering of the turbulent Prandlt number raises the overall peak temperature due to the enhanced thermal diffusion.

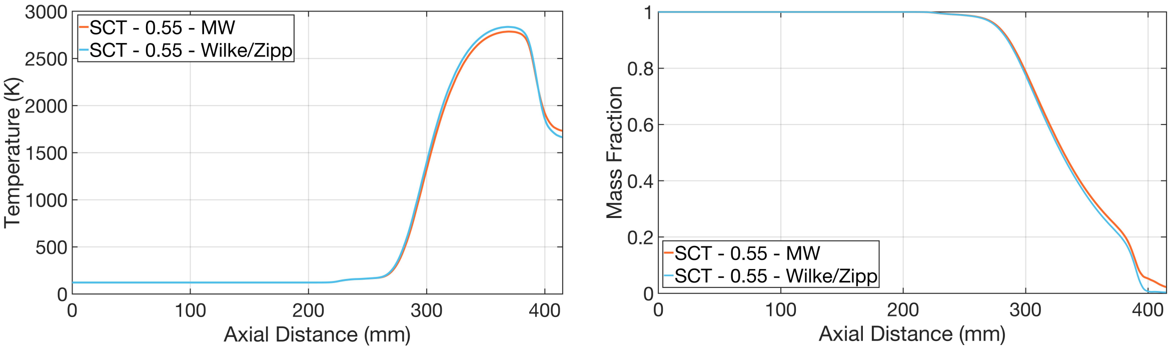


(A) Different turbulent Schmidt number axial temperature comparison

(B) Different turbulent Schmidt number axial O2 mass fraction comparison

FIGURE 7.10. Effects of lower turbulent Schmidt numbers on axial results

The effects of the Wilke and Zipperer/Herning mixing laws are presented in Fig.7.11. Minimal difference is observed as this region is highly turbulent. Unlike at the wall the effects of molecular transport are not dominant. One notable difference is the complete exhaustion of oxygen, with its mass fraction dropping to 0 before the nozzle throat at 400mm (light blue).



(A) Mass Weighted vs. Wilke and Zipperer/Herning mixing laws effect on axial temperature

(B) Mass Weighted vs. Wilke and Zipperer/Herning mixing laws effect on axial O2 mass fraction

FIGURE 7.11. Effects of different mixing laws on axial results

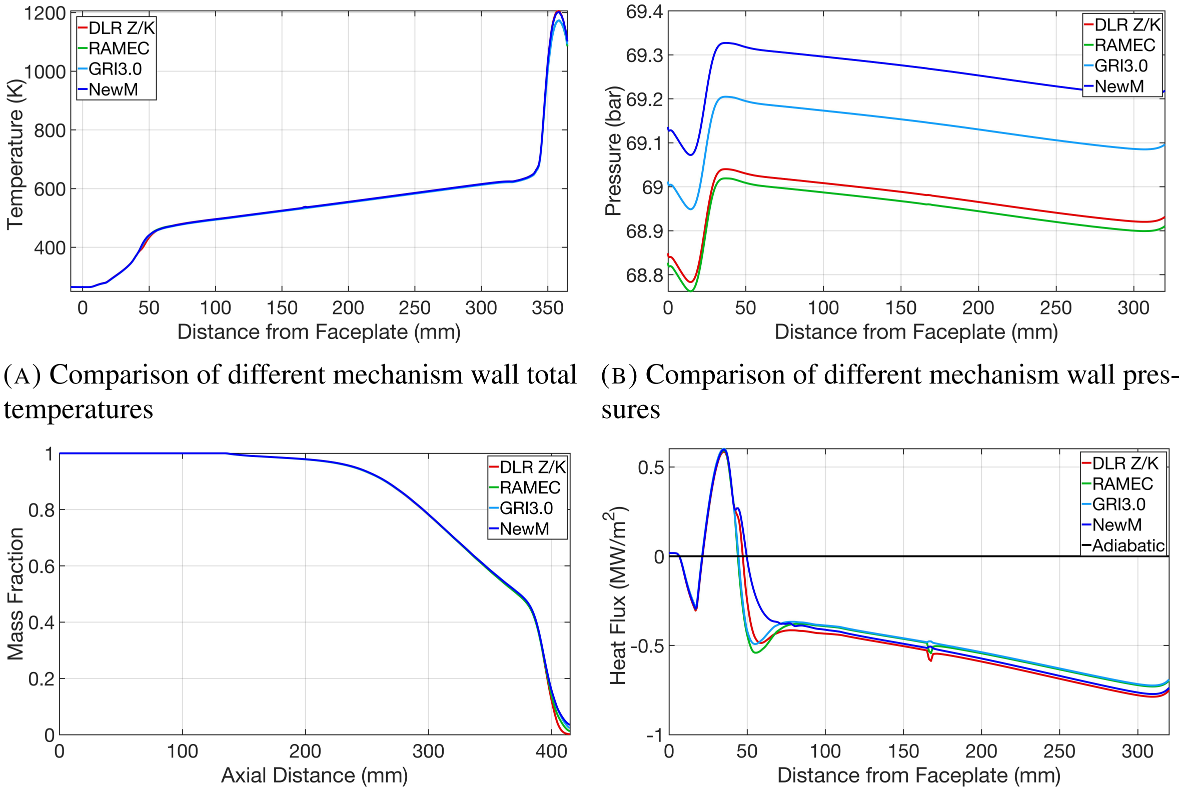
7.3 Chemical Mechanism Comparison

Fig.7.12 highlights the key differences between chemical mechanisms. The simulations were completed with the detailed flamelet and the embedded wall temperature profile (see Tab.7.5). Similar to the results found in literature, minor differences were noted. Wall temperature displayed no discernible difference, with a close inspection revealing a maximum 2 K difference in the 100-300 mm range. Similarly, O_2 saw a maximum 0.5 mm axial shift for a specific value within the 250-350 mm range. The pressures in Fig.7.12b varied by a maximum of 0.32 bar with the RAMEC scheme matching the Zhukov/Kong scheme the closest. Finally, the heat flux profiles followed similar trajectories. The NewM flux remained positive for slightly longer due to a small inflection in its downtrend at approximately 45 mm. Conversely a concavity shift is avoided 10 mm later, as seen in the other profiles. From 70 mm onwards the profiles match closely. The integrals are contained in Tab.7.6, rounded to the nearest kW. All schemes remained within a 3 kW range for the combustion chamber wall region from the faceplate to the start of the nozzle wall.

It is worth noting that the GRI3.0 has not been validated for pressures above 10 bar and temperatures below 1000 K. Whilst the RAMEC scheme has been validated at high pressures, it was designed for ramjet combustion with oxygen diluents, thus both these schemes have not been tested further. The NewM mechanism was explored further, verifying whether similar trends as seen in Fig.7.12 would occur with a more detailed simulation setup.

TABLE 7.5. Complete set of simulations

Simulation Name	LP	EoS	Turb. Model	Combustion Model	Chemical Mechanism	Transport Mixing	Transport Species	Wall Profile	Boundary Flow	Sct
DLR Z/K	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
NewM	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	NewM	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
GRI3.0	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	GRI3.0	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
RAMEC	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	RAMEC	None	Mixture Consant	Embedded	Standard Wall Functions	0.85
SCT - 0.70 - DLR Z/K	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	DLR Z/K	Mass Weighted	CEA	Embedded	Enhanced Wall Treatment	0.70
Final	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	NewM	Wilke/Zipp	CEA	Embedded	Enhanced Wall Treatment	0.70



(A) Comparison of different mechanism wall total temperatures

(B) Comparison of different mechanism wall pressures

(C) Comparison of different mechanism axial O2 mass fractions

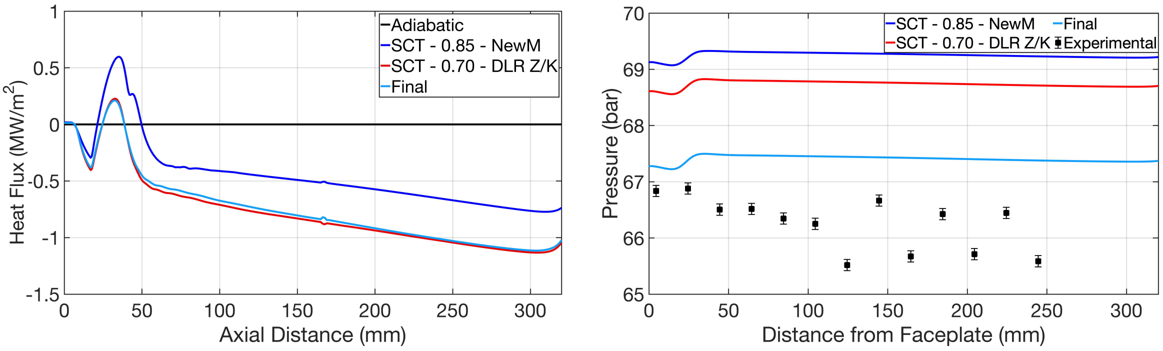
(D) Comparison of different mechanism wall heat fluxes

FIGURE 7.12. Different chemical mechanisms effect on wall pressure/temperature and axial O2 mass fraction and temperature

TABLE 7.6. Comparison of different mechanisms total heat transfer

Simulation	Total Heat Transfer (kW)
DLR Z/K	-24
RAMEC	-22
GRI3.0	-21
NewM	-22

A more complex simulation setup, marked final, was used to explore the effects of the NewM mechanism further (see Tab.7.5). The wall pressure and heat flux results are presented in Fig.7.13, yielding a divorce from previously observed trends. The wall heat flux out of the chamber in Fig.7.13a increases as expected, as the turbulent Schmidt number was lowered to 0.7 (light blue). Overlayed is the simulation that utilised a turbulent Schmidt number of 0.7 with the Zhukov/Kong mechanism (red). As seen in Fig.7.12d, the wall heat fluxes remain similar. No change in wall temperature was observed either. When looking at Fig.7.13b, a significant difference in pressure is observed. When using the NewM scheme with the detailed setup, a 0.6% deviation with the first experimental data point is noted. Decoupling between the wall heat flux and wall pressure has occurred, whilst also reversing the trend seen in Fig.7.12b which saw the NewM scheme raise the wall pressure. Further investigation is required to properly evaluate the cause of this behaviour, however it is likely that a differing species composition in the boundary flow has had an impact. The Wilke and Zipperer/Herning mixing laws capture a more realisitc species dependant viscosity and thermal conductivity, a key difference between simulations.



(A) NewM mechanism wall heat flux comparison

(B) NewM mechanism wall pressure comparison

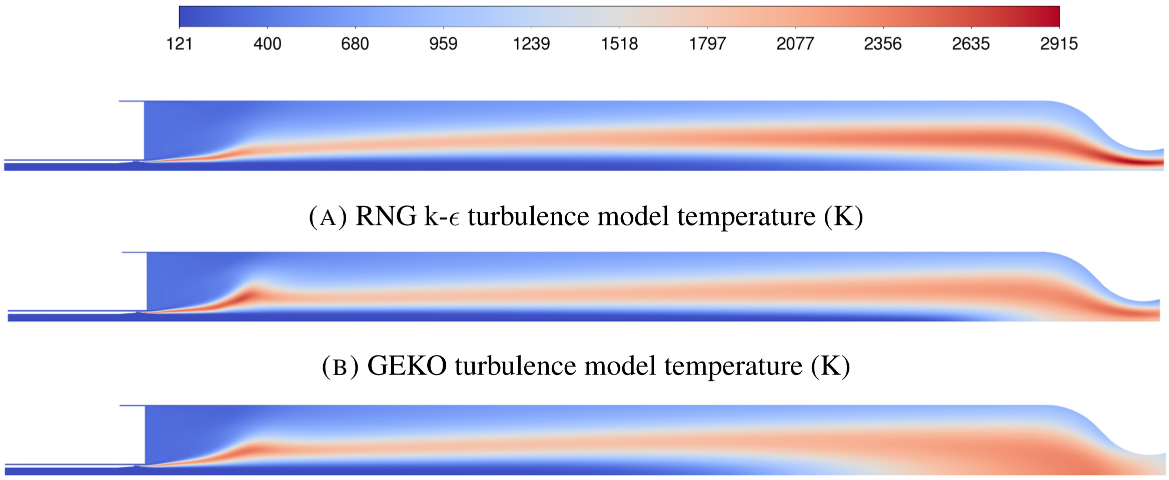
FIGURE 7.13. Effects of different simulation setups with the NewM mechanism

7.4 Turbulence Model Flow Field Contours

Through the testing of different turbulence models, significant differences in flame topology were observed. In all simulations utilising the standard k$-\epsilon$ model, the flame has been contained within the chamber with an exhausted LOX core. Whilst previous simulation setups have been analysed graphically along the walls and axis, their differences between contours are not easily discernible. Conversely, when using both the RNG and GEKO models, the flame shape changed dramatically, as seen below in Fig.7.14 and Fig.7.15. In both instances the flame extends out the nozzle alongside unburnt oxygen. Note, the GEKO contours refer to the simulation setup 1 in Tab.6.2, set to emulate the standard k-ϵ model. The effects of different parameters can be found in Appendix B. As the underlying mechanisms of this model are not disclosed, the results have been omitted from discussion here.

TABLE 7.7. Complete set of simulations

Simulation Name	LP	EoS	Turb. Model	Combustion Model	Chemical Mechanism	Transport Mixing	Transport Species	Wall Profile	Boundary Flow	Sct
Original	1	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
GEKO	1	SRK	GEKO	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Adiabatic	Standard Wall Functions	0.85
RNG	1	SRK	RNG	PDF-Flamelet New Profile	DLR Z/K	None	Mixture Consant	Adiabatic	Standard Wall Functions	0.85

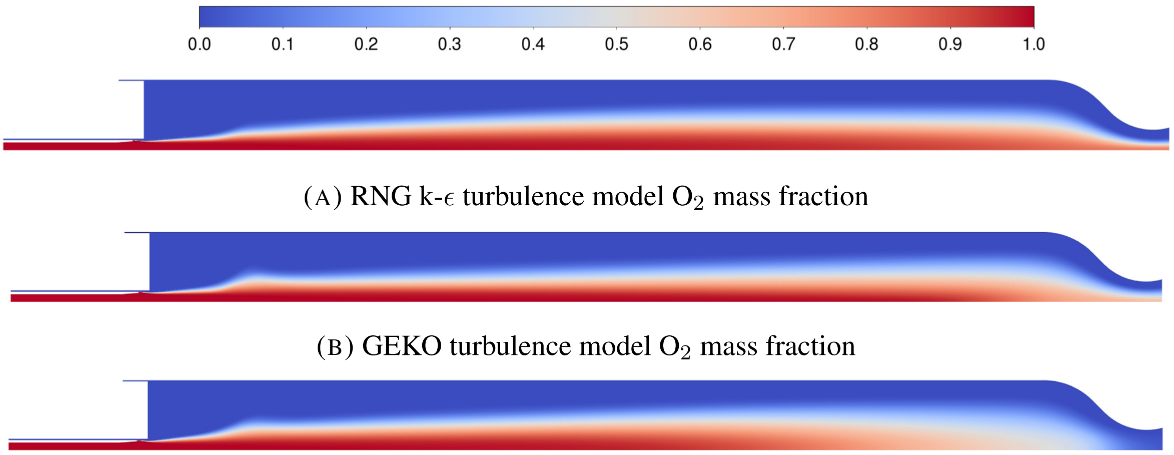


(A) RNG k-ϵ turbulence model temperature (K)

(B) GEKO turbulence model temperature (K)

(C) Original standard k-ϵ turbulence model temperature (K)

FIGURE 7.14. Comparison of turbulence model temperature contours



(A) RNG k-ϵ turbulence model O_2 mass fraction

(B) GEKO turbulence model O_2 mass fraction

(C) Original standard k-ϵ turbulence model O_2 mass fraction

FIGURE 7.15. Comparison of turbulence model O_2 mass fraction contours

Whilst both flames exited the nozzle, the RNG oxygen mass fraction remains above 0.8 along the axis paralleled by slow increase in temperature from its cryogenic injection. The hot shear layer that stems from the LOX post remains thin, reaching peak temperatures in the nozzle. The differential viscosity formulation (described in Sec.4.4.2), presents a key difference between the standard and RNG variations of the k-ϵ model. As this formulation aims to address the assumption that the flow is fully turbulent and incorporates low Reynolds effects more realistically, turbulent processes have been nullified.

The GEKO flame begins to tend back towards the axis, with larger drop in the mass fraction, however it still fails to close before the acceleration out the nozzle. The GEKO model is built from the k-ω model, thus has a different formulation at its core. These results mirror the work by Priyadarshini *et al.*, who tested variations of both k-ϵ and k-ω turbulence model classes, with only the standard k-ϵ conforming to the short flame produced experimentally in the G2 test case. Additional contours produced using the standard k-ϵ model are presented in Sec.7.5.

It is worth noting that RANS models cannot capture any inherent unsteadiness within the flow. Whilst the short flame produced with the standard k-ϵ model was initially assumed to be more accurate, there is the potential that the turbulent processes that facilitate the enhanced mixing and combustion underestimated the flame length. DLR's in house solver TAU also produced flames that are not contained within the chamber with the same geometry and injection load point. Whilst experimentally the flame length is not clear as it extends beyond the optically accessible region (see Fig.7.16), unsteadiness and lifted flames were observed (see Fig.3.11), presenting the potential for inaccuracy when comparing to the RANS steady state results [67]. Single injector configurations have no additional injection streams to mix which may cause longer flames. Further investigation is required experimentally and numerically with longer chambers to properly evaluate the steady state length of the flame.

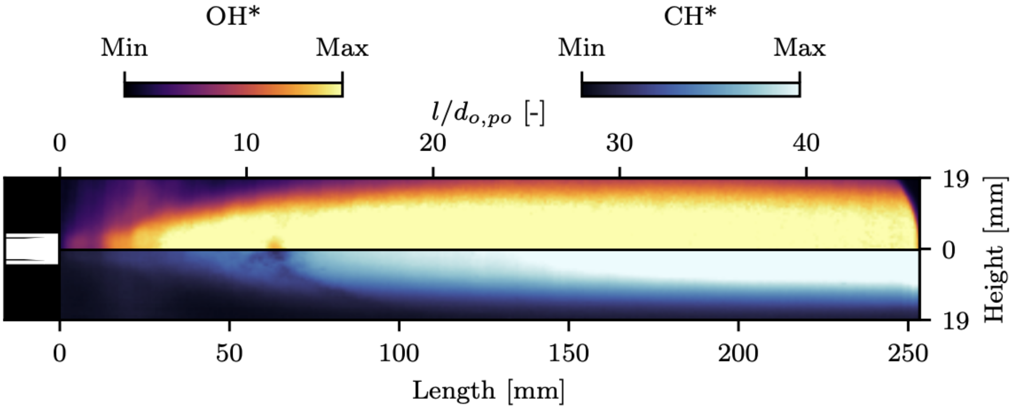


FIGURE 7.16. Time averaged OH* and CH* chemilumininescence captured through BKN's optical access window, taken from [67]

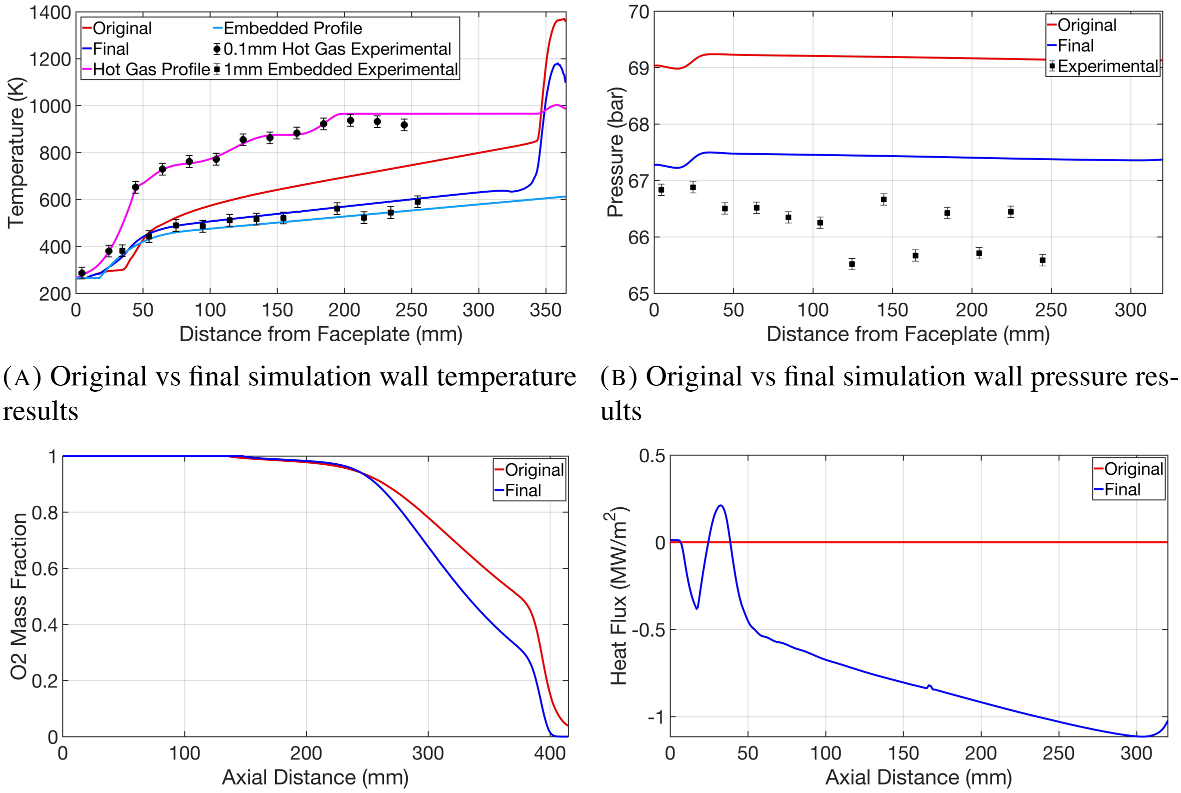
7.5 Final Simulation

7.5.1 Axial and Wall Results

The final simulation was run as a collation of all the modifications made from different areas of exploration. Any changes deemed to negatively effect the simulation were omitted. The complete setup is detailed in Tab.7.8, however the justification for the turbulent Schmidt number and mechanism is worth addressing. A turbulent Schmidt number of 0.7 was chosen over 0.55 and 0.85 as it formed a reasonable balance between improved pressure and heat flux without a large deviation in the axial position at which oxygen begins to burn. The NewM scheme was chosen for its additional detail in comparison to the Zhukov/Kong mechanism and the close conformation to experimental wall pressures.

TABLE 7.8. Original and final simulation setups

Simulation Name	LP	EoS	Turb. Model	Combustion Model	Chemical Mechanism	Transport Mixing	Transport Species	Wall Profile	Boundary Flow	Sct
Original	1	SRK	Sk-ϵ	PDF-Flamelet Log Profile	DLR Z/K	None	Mixture Constant	Adiabatic	Standard Wall Functions	0.85
Final	1	SRK	Sk-ϵ	PDF-Flamelet New Profile	NewM	Wilke/Zipp	CEA	Embedded	Enhanced Wall Treatment	0.70



(A) Original vs final simulation wall temperature results

(B) Original vs final simulation wall pressure results

(C) Original vs final simulation axial O2 mass fraction results

(D) Original vs final simulation wall heat flux results

FIGURE 7.17. Comparison between the original and final simulation setups with experimental data

Fig.7.17 presents four key results with respect to the original setup and experimental datapoints. The final wall temperature seen in Fig.7.17a (dark blue) remains below the original adiabatic values (red), aside from the first 50 mm due to its attachment to the embedded wall temperature profile. It was consistently observed that experimental wall temperatures rose quicker in the fore of the chamber. Whilst a large delay is observed with adiabatic temperatures (red), the calculated wall temperatures via the 1D conduction gradient (blue) reduce its deviation with experimental. A small overlap with the embedded wall profile (light blue) still remains between 25 mm and 40 mm corresponding to a brief moment of heat delivery to the chamber (see Fig.7.17d). Other modifications observed to influence the wall temperature non-negligibly were the implementation of CEA species transport modelling and lowering of the turbulent Schmidt number. Both of these changes raised the temperature by 40 K and 15 K respectively. Overall, the modelled wall thickness, material properties and conduction gradient method of calculating wall temperature has proved to be the most effective change.

Looking at the pressures in Fig.7.17b, a difference of 1.7 bar is observed. The final simulation (blue) conforms closely with the experimental measurements, yielding a 0.6% discrepancy with the first experimental data point. Small drops in pressure occurred when the embedded wall temperature profile, enhanced wall treatment, lower turbulent Schmidt number and complex mixing laws were applied sequentially. However the largest impact was made through the change in chemical mechanism to the NewM scheme. As discussed previously, this result was unexpected (Sec.7.3). Further investigation is required to verify the validity of the result, with a focus on the species composition at the wall.

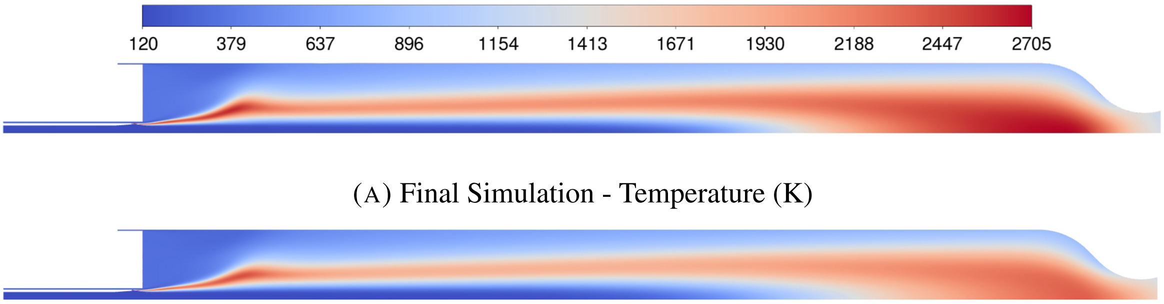
The oxygen mass fraction profiles, seen in Fig.7.17c remains similar until 250 mm. Here, the final simulation (blue) sees a faster oxygen burn with the mass fraction dropping to 0 at 400 mm. This trajectory is driven mainly by the turbulent Schmidt number difference, similar to that seen in Fig.7.10. Additionally, the complex transport mixing laws, have enforced a 0 mass fraction at the nozzle throat. As a result, the LOX core is completely exhausted and a shorter flame length is seen. Previously discussed modifications regarding boundary flow, near-wall treatment and transport modelling have negligible impacts here.

Finally, heat flux results are presented in Fig.7.17d. The original setup utilised adiabatic walls, hence not heat transfer is captured. A small moment of heat delivery to the chamber remains between 25 and 40 mm, discussed previously. The total heat transfer is -38 kW, producing a large discrepancy of 51.3% with experimental. When viewing the final wall pressure in Fig.7.17b it was expected that a larger total transfer of heat would be observed.

As discussed in Sec.7.3, the link between a lowered wall pressure and higher heat fluxes was broken when the Wilke and Zipperer/Herning mixing laws were implemented with the NewM scheme. Instead, the wall heat flux has remained constant. Given the lack of wall boundary conditions within the flamelet equations and no recombination reaction modelling, a wall heat flux deviation of 51.3% is justified. Addressing these simplifications or implementing the EDC method are two alternatives for future work to increase wall heat flux fidelity.

7.5.2 Flow Field Contours

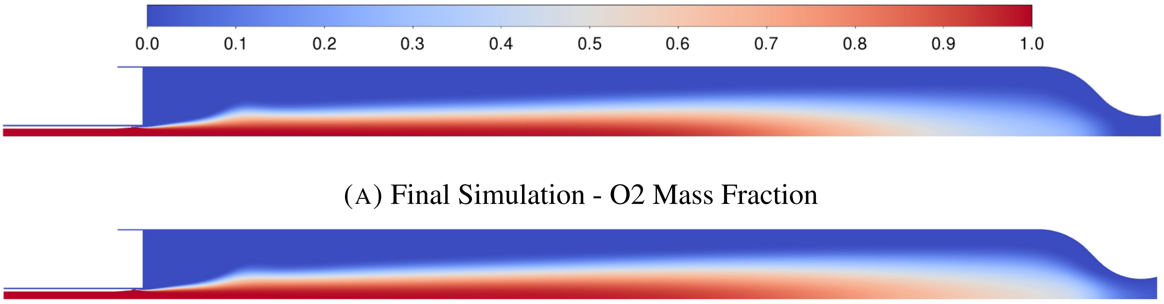
Fig.7.18 captures the flow field temperature of both the final and original simulation setups, indicative of the overall flame topology. The final solution produces a hotter and shorter flame, seen in Fig.7.18a, highlighting enhanced mixing and combustion. A thin layer sheared by the fuel and oxidiser jets expands away from the axis, with a steeper angle and higher temperature noted in comparison to the original solution. In the surveyed experimental setups, flame length and expansion angles were correlated with injection parameters such as the ROF, momentum flux ratio and recess of the LOX core. These parameters are identical between both simulations. Higher maximum temperatures captured in the flamelet tables, accompanied by intensified species/thermal diffusion processes induced by the lower turbulent Schmidt (and Prandtl) number can instead be noted as primary reasons for the observed difference. Fig.7.19 captures the difference in O_2 mass fraction, with a shorter LOX core observed in the final solution. This agrees with the comparison in axial mass fraction in Fig.7.17c.



(A) Final Simulation - Temperature (K)

(B) Original Simulation - Temperature (K)

FIGURE 7.18. Comparison between the original and final temperature contours



(A) Final Simulation - O2 Mass Fraction

(B) Original Simulation - O2 Mass Fraction

FIGURE 7.19. Comparison between the original and final O2 mass fraction contours

Fig.7.20 and Fig.7.21 capture the temperature dependant molecular viscosity and thermal conductivity respectively. Originally, constant values of 1.72×10^{-5} kg/ms and 4.54×10^{-2} W/mK were applied to the entire mixture, clearly unrealistic when observing the ranges yielded here. In particular, the thermal conductivity reaches values 2.5 times higher in the boundary flow, where its effect is dominant. Whilst both transport quantities rise with temperature, cryogenic liquid oxygen, modelled using NIST data also exudes high values. The supercritical injection causes rapid changes to thermo-physical properties such as specific heat and density (see Fig.3.8), seen in Fig.7.22 and Fig.7.23 respectively. The characteristic spike in specific heat as the injected oxygen crosses the widom line is observed in parallel with the steep drop in density, highlighting the importance of the SRK real gas EoS.

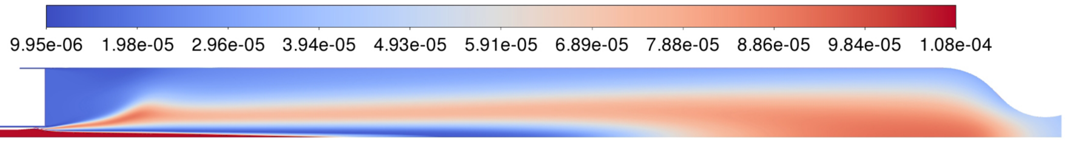


FIGURE 7.20. Molecular Viscosity (kg/ms)

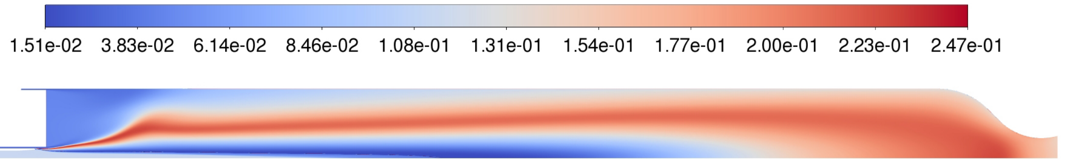


FIGURE 7.21. Thermal Conductivity (W/mK)

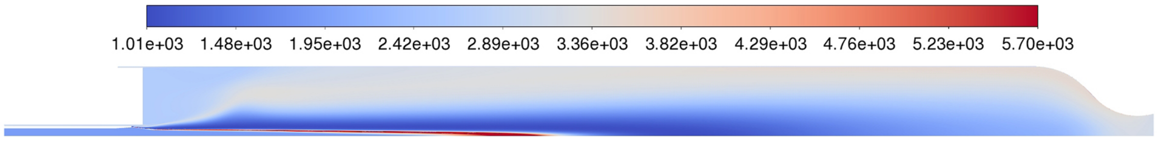


FIGURE 7.22. Specific Heat (J/kgK)

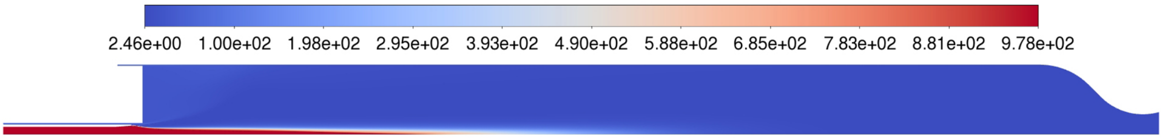


FIGURE 7.23. Density (kg/m^3)

In Sec.6.2.3, it was asserted that the actual range of scalar dissipation values would be required to evaluate the range captured in the flamelet tables. The NewM scheme predicted flame quenching beyond 610 000 1/s, therefore Fig.7.24 captures the scalar dissipation flow field trimmed to this maximum. This figure has been enlaraged at the LOX post, the only point in which the contour results exceed this value. Within the circular cutout, a peak scalar dissipation of 1 967 098 1/s is predicted. This value is 2.2% lower in comparison to the original setup with the DLR Zhukov/Kong mechanism, which had a peak of 2 010 374 1/s in the same region. Accordingly, the PDF-flamelet formulation predicts this region would correspond to an extinguished flame however it isn't explicitly modelled with the flow field solver instead retrieving values from the look-up tables at the maximum flamelet scalar dissipation. When using the NewM scheme, the maximum scalar dissipation extends 52.3% higher than the original extinction point of 400 000 when using the Zhukov/Kong mechanism. Whilst its important to note this increase in range and accuracy surrounding regions of high scalar dissipation, values in excess of 100 000 are rare within the scope of the whole chamber.

Finally, a direct comparison to the flame captured via BKN's optical access window is presented in Fig.7.25. The top image is the time averaged axisymmetric slice of OH* radiation taken experimentally. This is aligned with a numerical contour of OH molar concentration, mapped to the same colour scale. It is disclosed by Martin *et.al.* that the OH* has been

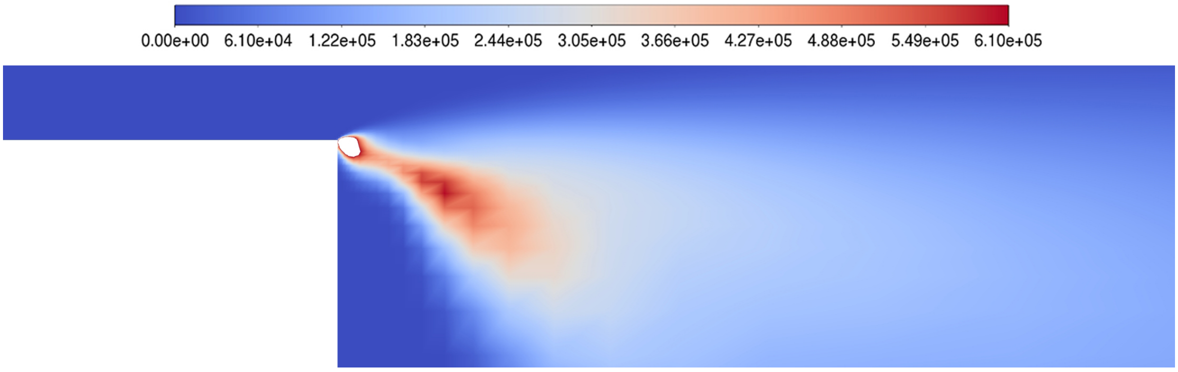


FIGURE 7.24. Scalar dissipation around the LOX post trimmed to the flamelet extinction point

amplified for higher contrast in the injector region, thus further downstream a uniform, abliet clipped, intensity is noted [67]. The exact relationship between OH* intensity and OH molar concentration is unknown, thus the numerical contour was mapped to the highest and lowest values seen in the equivalent optical access region. An important consideration is the physical 3D effects not captured in a 2D axisymmetric slice. The image captured through the optical access window is representative of complex radiation accumulation and absorption around the chamber. Whilst general topological observations can be made, furthered understanding of the radiation paths within the chamber should be pursued for precise comparison to numerical molar concentration.

As previously mentioned, the flame length extends beyond the 250mm range of the window, a result in agreement with the numerical study. Both flames expand gradually down the length of the chamber, with a quicker expansion seen experimentally. This is clearer in the fore of the chamber where a rapid increase in OH* intensity reaches a initial peak 25 mm in, marked by the dotted line. The same peak is reached numerically 10 mm further downstream. This agrees with the numerical results that noted a slower rise in temperatures along the wall. When profiles developed from experimental measurements are applied, this misalignment in the flame expansion peaks can be linked to the brief delivery of heat to the chamber. Whilst this may imply the current numerical result yields a flame that is still too long, further testing is required with optical access further downstream.

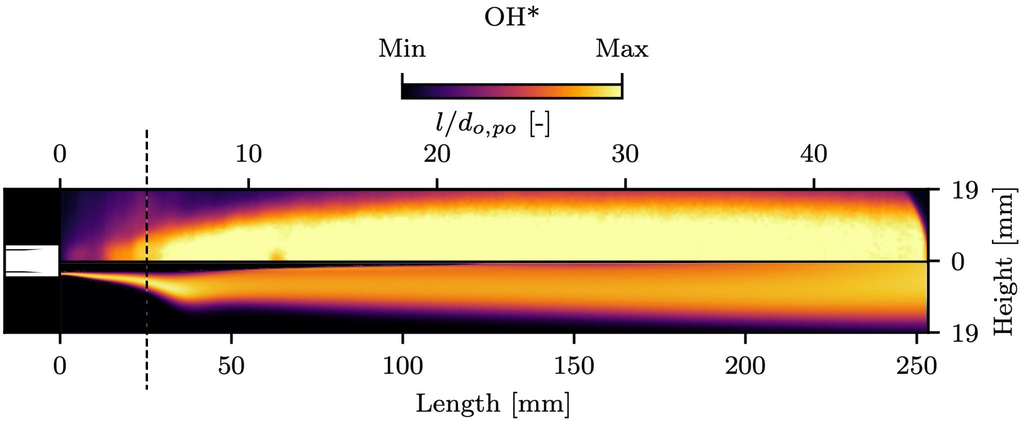


FIGURE 7.25. OH* chemiluminescence comparison (top) with numerical OH molar concentration (bottom)

CHAPTER 8

Conclusion

This chapter presents a summary of the thesis in Sec.8.1, with the future areas of work for the continuation of this project detailed in Sec.8.2.

8.1 Summary

This work has presented the RANS modelling of a single injector LOX/CH_4 research rocket combustion chamber, undertaken in conjunction with DLR at the Institute of Space Propulsion in Lampoldshausen Germany. This thesis aimed to produce a CFD simulation of higher fidelity, improving upon the groundwork established by Morris. Informed by an extensive survey of literature and Morris' baseline setup, six major areas of exploration were established. A sequential analysis through a series of evolving numerical calculations documented the effect of each change with the combined improvements collated into a final simulation.

The final setup utilised the PDF-flamelet method of modelling turbulent combustion, with a rigorous study of optimal flamelet parameters performed. Different chemical mechanisms were explored with the NewM scheme producing the closest agreement with experimental wall pressures. The standard k-ϵ turbulence model was used primarily, with the effect of internal parameters such as the turbulent Schmidt number and near-wall treatment explored. Lower turbulent Schmidt numbers were seen to intensify mixing and combustion, increasing temperatures and heat flux. A Schmidt number of 0.7 was chosen as it improved wall pressures and heat flux compared to 0.85, without pushing the starting point of axial oxygen mixing further downstream, as seen with 0.55. Enhanced wall treatment was used to resolve the boundary flow, proven to show the best conformity with experimental wall pressures as opposed to standard wall functions and the Menter Lechner formulation.

Additional turbulence models, namely the GEKO and RNG k-ϵ formulations yielded flames that extended out the nozzle, fueling further discussion surrounding the correct flame length. These results resembled the flames produced in TAU, adding to the possibility that the current flame length is too short. Further investigation both experimentally and numerically is required to properly define the flame length. Wall temperature profiles were tested, with a fixed profile fitted to thermocouples positioned 0.1 mm into the flow proving to be highly inaccurate, inducing heat delivery to the chamber. This was corrected using a 1D conduction gradient with a temperature profile applied behind the wall, incorporating wall material properties and thickness. Viscous transport quantities were partially modelled using the CEA and NIST database. In combination with the viscosity mixing law defined by Wilke, and the thermal conductivity mixing law defined by Zipperer/Herning, wall pressures and heat flux tended towards experimental results. The SRK EoS was chosen for all simulations, shown to effectively capture the rapid changes in oxygen's thermo-physical properties as it passes through the widom line and into the supercritical regime.

The relationships between different facets of the simulation setup and the validity of the modelling techniques used have been thoroughly documented. This research provides USYD with a detailed simulation test case and DLR with an alternative test case to compare with TAU. It also acts as source of comparison for future experimental studies. The continued investigation into the ambiguous flame length and large heat flux errors remains a priority, providing the starting point for a new placement and the continued investigation into the expanding field LOX/CH_4 combustion.

8.2 Future Outlook

8.2.1 Flame Length Investigation

The correct flame length remains unknown. There is also a discrepancy in the initial expansion angle, which may be the result of an incorrect flame length. Additionally, the total heat transfer is approximately half of the experimental measurement, a result that would be improved through a shorter flame. Conversely, the flames produced using the RNG and GEKO turbulence models and DLR's in-house solver TAU, imply that the flame length is underestimated. The unsteadiness observed experimentally also implies that its comparison with a RANS simulation may be invalid.

In order to address this problem, continued numerical and experimental studies are required. Extending the combustion chamber with optical access that captures the end of a stable flame should be be achieved experimentally. The same extension should be made to the mesh in the numerical setup. This allows for the furthered study of turbulent models that produce longer flames with a direct comparison to TAU. The relationship between OH* radiation and numerical OH molar concentration/mass fraction slices should also be studied for a more informed comparison. Other model simplifications may explain discrepancies in heat flux and expansion angles, discussed below.

8.2.2 Improved Transport Modelling

In order to incorporate laminar viscosity and thermal conductivity, an algorithm was written than imports the CEA database and a chosen chemical mechanism and exports individual UDF's for each species. If a species from the mechanism isn't found in the CEA database, no UDF is produced and the value in Fluent is left as the default constant. Within the timeframe of this thesis it wasn't deemed a priority to address this, however a formulation for species not in the CEA database would be of value. The Lennard-Jones method could be explored, utilising the parameters provided in the transport data file for each mechanism. An extension to the written function should be made so that it also imports the transport file. In the case a species is not found, it references this file and produces a new UDF using the Lennard-Jones method.

8.2.3 GEKO Turbulence Model

Currently the GEKO turbulence model lacks the mathematical basis to justify its use. As it is a new release, derivations for the free parameters can be expected sometime in the future. Once there is an understanding of its internal mechanisms a rigorous exploration of its parameters should be made for a valid comparison against other models. This investigation would be of greater value in conjunction with the aforementioned extended chamber geometry to properly evaluate the flame length.

8.2.4 Eddy Dissipation Concept

The PDF-flamelet method excels in computational efficiency however its accuracy is inferior in comparison to full finite rate models such as the EDC. It was briefly attempted in this work however it was abandoned due to its complexity and calculation time. With access to a high-powered desktop or a cluster, it would be feasible to implement the EDC and explore its return on accuracy in comparison to the PDF-flamelet method. A focused study on the wall heat flux should be made, as the current discrepancy with experimental is predicted to be a result of the flamelet model simplifications. Additionally, flame expansion angles and the rise in temperature along the wall should be explored. Alternatively, a more detailed flamelet model with incorporated real gas effects and flame-wall interactions could be explored. Again, this in conjunction with different geometries and updated experimental studies would assist in defining a more conclusive flame length.

8.2.5 Wall Conduction

The addition of wall thickness and material properties alongside a temperature profile behind the wall drastically improved wall temperatures. As the wall temperature is calculated using simple 1D conduction, a more complex model that utilises a more realistic multi-dimensional transfer of heat would be of interest. Currently heat flux results are too small, with simplified conduction believed to be a contributing factor.

8.2.6 Chemical Mechanisms

Differing trends were seen between mechanisms, namely with the NewM wall pressures. With a simple simulation setup close to the original, the NewM scheme produced higher pressures in comparison to the Zhukov/Kong mechanism. The opposite occurred in the final simulation, with the NewM scheme dropping the wall pressure further than any other single simulation change. Overall, different schemes have effected the results more than expected, without clear relationships. A deeper investigation into their effects in conjunction with other facets of the simulation would be valuable.

Bibliography

[1] URL: http://combustion.berkeley.edu/gri-mech/version30/text30.html.

[2] URL: https://www.hightempmetals.com/techdata/hitempInconel600data.php.

[3] V Ahuja et al. ‘Simulations of Transient Phenomena in Liquid Rocket Feed Systems’. In: *NASA Technical Reports Server* (Jan. 2006).

[4] David Altman. ‘Rocket Motors, Hybrid’. In: *Encyclopedia of Physical Science and Technology (Third Edition)*. Ed. by Robert A. Meyers. Third Edition. New York: Academic Press, 2003, pp. 315–338. ISBN: 978-0-12-227410-7. DOI: https://doi.org/10.1016/B0-12-227410-5/00123-X. URL: https://www.sciencedirect.com/science/article/pii/B012227410500123X.

[5] Redha Amri, D. Gibbon and T. Rezoug. ‘The design, development and test of one newton hydrogen peroxide monopropellant thruster’. In: *Aerospace Science and Technology* 25.1 (2013), pp. 266–272. ISSN: 1270-9638. DOI: https://doi.org/10.1016/j.ast.2012.02.002. URL: https://www.sciencedirect.com/science/article/pii/S1270963812000259.

[6] Wolfgang Armbruster et al. ‘Injector-Driven Flame Dynamics in a High-Pressure Multi-Element Oxygen–Hydrogen Rocket Thrust Chamber’. In: *Journal of Propulsion and Power* 35.3 (2019), pp. 632–644. DOI: 10.2514/1.B37406. eprint: https://doi.org/10.2514/1.B37406. URL: https://doi.org/10.2514/1.B37406.

[7] Richard ARNOLD, Dmitry I. SUSLOV and Oskar J. HAIDN. ‘Experimental investigation of film cooling with tangential slot injection in a LOX/CH4 subscale rocket

combustion chamber'. In: *TRANSACTIONS OF THE JAPAN SOCIETY FOR AERONAUTICAL AND SPACE SCIENCES, SPACE TECHNOLOGY JAPAN* 7.ists26 (2009). DOI: 10.2322/tstj.7.pa_81.

[8] Richard Blockley, Wei Shyy and James L Cannon. 'Propulsion and Power'. In: *Encyclopedia of Aerospace Engineering*. John Wiley amp; Sons, 2010.

[9] *Brief history of rockets*. URL: https://www.grc.nasa.gov/WWW/K-12/TRC/Rockets/history_of_rockets.html.

[10] Luz Calle and Louis MacDowell. '35 Years of Corrosion Protection at the Kennedy Space Center'. In: (Apr. 2022).

[11] *Cassini Completes Its Historic Journey with Plunge into Atmosphere of Saturn*. Sept. 2017. URL: http://www.sci-news.com/space/cassini-historic-journey-05230.html.

[12] B. Chehroudi, D. Talley and E. Coy. 'Visual characteristics and initial growth rates of round cryogenic jets at subcritical and supercritical pressures'. In: *Physics of Fluids* 14.2 (2002), pp. 850–861. DOI: 10.1063/1.1430735. eprint: https://doi.org/10.1063/1.1430735. URL: https://doi.org/10.1063/1.1430735.

[13] *Chemical Propulsion Systems - Glenn Research Center*. June 2021. URL: https://www1.grc.nasa.gov/research-and-engineering/chemical-propulsion-systems/.

[14] Alexander Chemnitz. 'Analysis and Improvement of Rocket Engine Combustion Stability Simulations'. In: (Oct. 2021).

[15] Alexander Chemnitz et al. 'Numerical modeling of flow and combustion in a single-element GCH4/GOX rocket combustor: Aspects of Turbulence modeling'. In: *52nd AIAA/SAE/ASEE Joint Propulsion Conference* (2016). DOI: 10.2514/6.2016-4998.

[16] John D Clark and Isaac Asimov. *Ignition!* Rutgers University Press, 2021.

[17] *Cryogenics*. URL: https://www.britannica.com/science/cryogenics.

[18] Luigi Cutrone, Francesco Battista and Giuliano Ranuzzi. 'Supercritical High Pressure Combustion Simulation for LOX/CH4 Rocket Propulsion Systems'. In: July 2008. DOI: 10.2514/MJPC08.

[19] Rainer N. Dahms and Joseph C. Oefelein. 'Atomization and dense-fluid breakup regimes in liquid rocket engines'. In: *Journal of Propulsion and Power* 31.5 (2015), pp. 1221–1231. DOI: 10.2514/1.b35562.

[20] P. A. Davidson. *Turbulence: An introduction for scientists and Engineers*. Oxford University Press, 2015.

[21] Maria Grazia De Giorgi, Aldebara Sciolti and Antonio Ficarella. 'Application and Comparison of Different Combustion Models of High Pressure LOX/CH4 Jet Flames'. In: *Energies* 7 (Jan. 2014), pp. 477–497. DOI: 10.3390/en7010477.

[22] Maria Grazia De Giorgi, Aldebara Sciolti and Antonio Ficarella. 'Comparisons between different combustion models for High pressure LOX/ CH4 jet flames'. In: June 2011. DOI: 10.2514/6.2011-3587.

[23] D. Dicara and Jose Gonzalez del Amo. 'Electric Propulsion Technology Overview'. In: June 2017.

[24] *DLR - über den Standort Lampolshausen*. URL: https://www.dlr.de/content/de/artikel/standorte/lampoldshausen/ueber-lampoldshausen.html.

[25] *DLR – DLR entwickelt Neue Forschungsbrennkammer*. URL: https://www.dlr.de/content/de/artikel/news/2021/01/20210311_blick-ins-feuer.html.

[26] *DLR site lampoldshausen an overview*. URL: https://www.dlr.de/content/en/downloads/2018/standortflyer-dlr-lampoldshausen_3061.pdf?__blob=publicationFile&amp;v=14.

[27] Tim Dodd. *Is spacex's Raptor engine the King of Rocket Engines?* Dec. 2020. URL: https://everydayastronaut.com/raptor-engine/.

[28] Mark L. Dranovskij and Vigor Yang. *Combustion instabilities in liquid rocket engines: Testing and development practices in Russia*. American Inst. of Aeronautics and Astronautics, 2007.

[29] Brian Dunbar. *The F-1 engine powered Apollo into history, Blazes Path for space launch system advanced propulsion*. Mar. 2016. URL: https://www.nasa.gov/topics/history/features/f1_engine.html.

[30] Van Huff N E. and David A. Fairchild. *Liquid rocket engine fluid-cooled combustion chambers*. National Aeronautics and Space Administration, 1972.

[31] The Editors of Encyclopaedia Britannica. *Eugen Sänger*. URL: https://www.britannica.com/biography/Eugen-Sanger.

[32] E F Fiock. 'High Speed Aerodynamics and Jet Propulsion'. In: *Physical measurements in gas dynamics and Combustion*. Princeton University Press, 1954.

[33] Gabriela Costa Pinto Fiuza and André Luiz Tenório Rezende. 'Comparison of K-E Turbulence Model Wall Functions Applied on a T-Junction Channel Flow'. In: *International Journal of Engineering Research and* 4 (2018).

[34] Jeff Foust. *NASA completes final test of SLS boosters before First Launch*. June 2016. URL: https://spacenews.com/nasa-completes-final-test-of-sls-boosters-before-first-launch/.

[35] Jonathan H Frank and Robert S Barlow. *Non-premixed Turbulent Combustion*. Sandia National Laboratories, 2007.

[36] Alessio Frassoldati et al. 'Simplified kinetic schemes for oxy-fuel combustion'. In: (Jan. 2009).

[37] Duncan French et al. 'Numerical Analysis of Chamber Wall Heat Fluxes in a LOX/CH4 Single Injector Rocket'. In: July 2015. DOI: 10.2514/6.2015-3758.

[38] 'GEKO – A New Paradigm in Turbulence Modeling'. In: *ANSYS White Paper* (2021).

[39] *Geko – a new paradigm in turbulence modeling | ansys white paper*. URL: https://www.ansys.com/resource-center/white-paper/geko-turbulence-modeling.

[40] *Germany Smart Traveller*. URL: https://www.smartraveller.gov.au/destinations/europe/germany.

[41] M. HABIBALLAH* et al. 'Experimental studies of high-pressure cryogenic flames on the Mascotte Facility'. In: *Combustion Science and Technology* 178.1-3 (2006), pp. 101–128. DOI: 10.1080/00102200500294486.

[42] Jennifer Harbaugh. *Space launch system solid rocket booster*. Feb. 2020. URL: https://www.nasa.gov/exploration/systems/sls/fs/solid-rocket-booster.html.

[43] Justin Hardi et al. 'High frequency combustion instabilities in liquid propellant rocket engines: research programme at DLR Lampoldshausen'. In: Jan. 2016.

[44] Jean-Pierre Hickey and Matthias Ihme. 'Supercritical mixing and combustion in rocket propulsion'. In: Jan. 2014.

[45] Dieter K Huzel. *Modern engineering for design of liquid-propellant rocket engines*. Vol. 147. AiAA, 1992.

[46] Won-Sub Hwang et al. 'Numerical simulation of a GH2/LOX single injector combustor and the effect of the turbulent Schmidt number'. In: *Energies* 13.24 (2020), p. 6616. DOI: 10.3390/en13246616.

[47] Alessandro de Iaco Veris. *Fundamental Concepts of Liquid-Propellant Rocket Engines*. Springer International Publishing, 2020. ISBN: 9783030547042.

[48] ANSYS Inc. 'ANSYS Fluent Customisation Guide'. In: (2021).

[49] ANSYS Inc. 'ANSYS Fluent Theory Guide'. In: (2021).

[50] ANSYS Inc. 'ANSYS Fluent User Guide'. In: (2021).

[51] *Is real gas flamelet generation possible?* URL: https://forum.ansys.com/discussion/33663/is-real-gas-flamelet-generation-possible.

[52] B. Ivancic et al. 'Investigation of different modeling approaches for CFD simulation of high pressure rocket combustors'. In: Jan. 2013, pp. 1–15. DOI: 10.1051/eucass/201608095.

[53] David Krejci and Paulo Lozano. 'Space Propulsion Technology for Small Spacecraft'. In: *Proceedings of the IEEE* 106.3 (2018), pp. 362–378. DOI: 10.1109/JPROC.2017.2778747.

[54] Turner Martin J L. *Rocket and spacecraft propulsion: Principles, practice and new developments*. Springer published in association with Praxis Pub., 2010.

[55] Keith J. Laidler. *Chemical Kinetics*. HarperCollins, 1987.

[56] Chung K. Law. *Combustion Physics*. Cambridge University Press, 2006. DOI: 10.1017/CBO9780511754517.

[57] P H Lebrun. ‘AN INTRODUCTION TO CRYOGENICS’. In: *EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH* (Jan. 2007).

[58] Justin Locke and Roger Woodward. ‘Chamber Wall Heat Flux Measurements for a LOX/CH4 Uni-element Rocket’. In: *Collection of Technical Papers - 43rd AIAA/ASME/SAE/ASEE Joint Propulsion Conference* 5 (July 2007). DOI: 10.2514/6.2007-5547.

[59] Johannes Lux and Oskar Haidn. ‘Effect of Recess in High-Pressure Liquid Oxygen/-Methane Coaxial Injection and Combustion’. In: *Journal of Propulsion and Power* 25.1 (2009), pp. 24–32. DOI: 10.2514/1.37308.

[60] Johannes Lux and Oskar Haidn. ‘Flame Stabilization in High-Pressure Liquid Oxygen/Methane Rocket Engine Combustion’. In: *Journal of Propulsion and Power* Vol. 25 (Jan. 2009). DOI: 10.2514/1.36852.

[61] Peter Ma et al. ‘Nonadiabatic Flamelet Formulation for Predicting Wall Heat Transfer in Rocket Engines’. In: *AIAA Journal* 56 (Apr. 2018), pp. 1–14. DOI: 10.2514/1.J056539.

[62] Leandro Magalhães et al. ‘Turbulence Modeling Insights into Supercritical Nitrogen Mixing Layers’. In: *Energies* 13 (Apr. 2020). DOI: 10.3390/en13071586.

[63] B. MAGNUSSEN. ‘On the structure of turbulence and a generalized eddy dissipation concept for chemical reaction in turbulent flow’. In: *19th Aerospace Sciences Meeting* (1981). DOI: 10.2514/6.1981-42.

[64] Amir Mardani, Arash Ghasempour and Mohammad Farshchi. ‘Numerical Investigation of Gaseous Hydrogen and Liquid Oxygen Combustion under Subcritical Condition’. In: *Energy Fuels* 33 (Aug. 2019). DOI: 10.1021/acs.energyfuels.9b02050.

[65] *Mars Rover Perseverance in Safe Mode: What Does It Mean?* Aug. 2020. URL: https://techtheday.com/mars-rover-perseverance-in-safe-mode-what-does-it-mean/.

[66] Jan Martin et al. ‘Experimental Investigation of Self-Excited Combustion Instabilities in a LOX/LNG Rocket Combustor’. In: *Journal of Propulsion and Power* 37.6 (2021),

pp. 944–951. DOI: 10.2514/1.B38289. eprint: https://doi.org/10.2514/1.B38289. URL: https://doi.org/10.2514/1.B38289.

[67] Jan Martin et al. ‘Flame characteristics and response of a high-pressure LOX/CNG rocket combustor with large optical access’. In: (2021).

[68] Jan Martin et al. ‘Flame characteristics of a high-pressure LOX/H2 rocket combustor with large optical access’. In: *Case Studies in Thermal Engineering* 28 (Oct. 2021), p. 101546. DOI: 10.1016/j.csite.2021.101546.

[69] Elisei Maslov. *Detailed diagram of the SpaceX raptor engine*. Aug. 2019. URL: https://twitter.com/eliseimaslov/status/1167889621415415809?lang=en.

[70] Wolfgang Mayer and Hiroshi Tamura. ‘Propellant injection in a liquid oxygen/gaseous hydrogen rocket engine’. In: *Journal of Propulsion and Power* 12.6 (1996), pp. 1137–1147. DOI: 10.2514/3.24154.

[71] Wolfgang O. H. Mayer et al. ‘Atomization and Breakup of Cryogenic Propellants Under High-Pressure Subcritical and Supercritical Conditions’. In: *Journal of Propulsion and Power* 14.5 (1998), pp. 835–842. DOI: 10.2514/2.5348. eprint: https://doi.org/10.2514/2.5348. URL: https://doi.org/10.2514/2.5348.

[72] Ryan McLaren and M. Ragheb. ‘Nuclear propulsion choices for space exploration’. In: Apr. 2010, pp. 1–7. DOI: 10.1109/INREC.2010.5462568.

[73] Bart Merci, D. Roekaerts and Amsini Sadiki. *Experiments and numerical simulations of diluted spray turbulent combustion: Proceedings of the 1st International Workshop on Turbulent Spray Combustion*. Springer, 2011.

[74] Are Mjaavatten. *polyfix(x,y,n,xfix,yfix,xder,dydx)*. 2020. URL: https://au.mathworks.com/matlabcentral/fileexchange/54207-polyfix-x-y-n-xfix-yfix-xder-dydx.

[75] Clara Morris. ‘RANS Modelling of Supercritical LOX/CH4 Combustion in DLR’s BKN Rocket Combustion Chamber’. In: (2022).

[76] *NIST Chemistry Webbook*. URL: https://webbook.nist.gov/chemistry/.

[77] Joseph C. Oefelein. ‘Thermophysical characteristics of shear-coaxial LOX–H2 flames at supercritical pressure’. English. In: *Proceedings of the Combustion Institute* 30.2 (2005), pp. 2929–2937. DOI: `10.1016/j.proci.2004.08.212`.

[78] M. Oschwald and A. Schik. ‘Supercritical nitrogen free jet investigated by spontaneous Raman scattering’. In: *Experiments in Fluids* 27.6 (1999), pp. 497–506. DOI: `10.1007/s003480050374`.

[79] Stephanie Pohl et al. ‘Real gas CFD simulations of hydrogen/oxygen supercritical combustion’. In: *EUCASS Proceedings Series* 4 (Mar. 2013), pp. 583–614. DOI: `10.1051/eucass/201304583`.

[80] Thierry Poinsot and D. Veynante. *Theoretical and numerical combustion*. T. Poinsot, D. Veynante, 2012.

[81] Sindhuja Priyadarshini et al. ‘Numerical investigation of coaxial GCH4/lox combustion at supercritical pressures’. In: *Combustion Science and Technology* 193.11 (2020), pp. 1973–1997. DOI: `10.1080/00102202.2020.1723009`.

[82] G. Ribert et al. ‘Simulation of supercritical flows in rocket-motor engines: Application to Cooling Channel and injection system’. In: *Progress in Propulsion Physics* (2013). DOI: `10.1051/eucass/201304205`.

[83] R.L. Sackheim. ‘Spacecraft Chemical Propulsion’. In: *Encyclopedia of Physical Science and Technology (Third Edition)*. Ed. by Robert A. Meyers. Third Edition. New York: Academic Press, 2003, pp. 315–338. ISBN: 978-0-12-227410-7. DOI: `https://doi.org/10.1016/B0-12-227410-5/00123-X`. URL: `https://www.sciencedirect.com/science/article/pii/B012227410500123X`.

[84] Moritz Schulze and Thomas Sattelmayer. ‘Linear Stability Assessment of a cryogenic rocket engine’. In: *International Journal of Spray and Combustion Dynamics* 9.4 (2017), pp. 277–298. DOI: `10.1177/1756827717695281`.

[85] Aldebara Sciolti. ‘Different Combustion Models Applied to High Pressure LOX/CH4 Jet Flames’. In: June 2011.

[86] Maziar Shafaee et al. ‘Numerical investigation of propellant flow and finite element analysis of wall structure for a bi-propellant thruster, compared to proposed analytical

results'. In: *International Journal of Mechanical Engineering and Robotics Research.* (Apr. 2016). DOI: 10.18178/ijmerr.5.2.133-137.

[87] Tausif Shaikh, Lalit Patidar and Arindrajit Chowdhury. 'Experimental and Numerical Investigation of Combustion in a Hydrocarbon and Gaseous Oxygen fuelled Rocket'. In: *Applied Thermal Engineering* 110 (Sept. 2016). DOI: 10.1016/j.applthermaleng.2016.08.222.

[88] Abhishek Sharma and S. Sunil Kumar. 'NUMERICAL MODELLING OF SUPER-CRITICAL COMBUSTION IN LOX/METHANE MULTI ELEMENT CHAMBER'. In: Oct. 2018.

[89] Aditya Shukla, Jignesh Vaghasia and Manish Mistry. 'Effect of laser ignition on combustion and performance of internal combustion engine: A Review'. In: *Energy Conversion and Management: X* 13 (2022), p. 100166. ISSN: 2590-1745. DOI: https://doi.org/10.1016/j.ecmx.2021.100166. URL: https://www.sciencedirect.com/science/article/pii/S259017452100091X.

[90] Ghislain Singla et al. 'Transcritical oxygen/transcritical or supercritical methane combustion'. In: *Proceedings of the Combustion Institute* 30 (Jan. 2005), pp. 2921–2928. DOI: 10.1016/j.proci.2004.08.063.

[91] Giorgio Soave. 'Equilibrium Constants From a Modified Redlich-Kwong Equation of State'. In: *Chemical Engineering Science - CHEM ENG SCI* 27 (June 1972), pp. 1197–1203. DOI: 10.1016/0009-2509(72)80096-4.

[92] Sebastian Soller et al. 'Characterisation of a LOX-LCH4 Gas Generator'. In: June 2014.

[93] Sebastian Soller et al. 'Combustion Stability Characteristics of Coax-Swirl-Injectors for Oxygen/Kerosene'. In: vol. 6. July 2007. ISBN: 978-1-62410-011-6. DOI: 10.2514/6.2007-5563.

[94] Evgeny Strokach et al. 'Simulation of a gox-GCH4 rocket combustor and the effect of the geko turbulence model coefficients'. In: *Aerospace* 8.11 (2021), p. 341. DOI: 10.3390/aerospace8110341.

[95] George P. Sutton and Oscar Biblarz. *Rocket Propulsion Elements*. John Wiley amp; Sons Inc., 2017.

[96] Roger Svelha. 'Transport Coefficients for the NASA Chemical Equilibrium'. In: *NASA Technical Memorandum 4647* (Apr. 1995).

[97] M Théron et al. 'Experimental and numerical investigation of LOX/Methane Cryogenic Combustion at low mixture ratio'. In: July 2019.

[98] Hilda Vernin and Pascal Pempie. 'LOX/CH4 and LOX/LH2 Heavy Launch Vehicle Comparison'. In: Aug. 2009. ISBN: 978-1-60086-972-3. DOI: 10.2514/6.2009-5133.

[99] *Vulcain engine*. URL: https://www.esa.int/Enabling_Support/Space_Transportation/Launch_vehicles/Vulcain_engine.

[100] Zhen-Guo Wang. *Internal combustion processes of liquid rocket engines: modeling and numerical simulations*. John Wiley & Sons, 2016.

[101] C. R. Wilke. 'A viscosity equation for gas mixtures'. In: *The Journal of Chemical Physics* 18.4 (1950), pp. 517–519. DOI: 10.1063/1.1747673.

[102] F.A. Williams. 'Combustion'. In: *Encyclopedia of Physical Science and Technology (Third Edition)*. Ed. by Robert A. Meyers. Third Edition. New York: Academic Press, 2003, pp. 315–338. ISBN: 978-0-12-227410-7. DOI: https://doi.org/10.1016/B0-12-227410-5/00123-X. URL: https://www.sciencedirect.com/science/article/pii/B012227410500123X.

[103] Matt Williams. *Neil Armstrong: The first man to walk on the Moon*. Dec. 2015. URL: https://www.universetoday.com/50337/neil-armstrong-facts/.

[104] M. Wolfshtein. 'The velocity and temperature distribution in one-dimensional flow with turbulence augmentation and pressure gradient'. In: *International Journal of Heat and Mass Transfer* 12.3 (1969), pp. 301–318. DOI: 10.1016/0017-9310(69)90012-x.

[105] Baoe Yang, Francesco Cuoco and Michael Oschwald. 'Atomization and flames in LOX/H2- and LOX/CH4- Spray Combustion'. In: *Journal of Propulsion and Power* 23.4 (2007), pp. 763–771. DOI: 10.2514/1.26538.

[106] Hong Ye et al. 'Numerical simulation and measurements of wall heat fluxes in a single-element GO2/GCH4 rocket combustor'. In: *EUCASS2017* (2017). DOI: 10.13009.

[107] Victor Zhukov and Alan Kong. ‘Development of Skeletal Kinetic Mechanism of Methane Oxidation for High Pressures and Temperatures’. In: Jan. 2016.

[108] Victor P. Zhukov. ‘Computational Fluid Dynamics Simulations of a GO2/GH2 Single Element Combustor’. In: *Journal of Propulsion and Power* 31.6 (2015), pp. 1707–1714. DOI: 10.2514/1.B35654. eprint: https://doi.org/10.2514/1.B35654. URL: https://doi.org/10.2514/1.B35654.

[109] L. Zipperer and F. Herning. ‘Beitrag zur Berechnung der Zähigkeit technischer Gasgemische aus den Zähigkeitswerten der Einzelbestandteile’. In: *Das Gas- und Wasserfach* 4 (1936).

APPENDIX A

Transport UDF Generation

A1 CEA Transport UDF Algorithm

The script is modularised by individual functions that break up the overarching steps defined in Chapter 6. The inbuilt MATLAB function `fileread` is used to read the CEA database and chemical mechanism text files that are stored in the local directory. MATLAB stores these files as character arrays from which the required data needs to be extracted. The function `strfind` is used to locate strings within the char arrays and return an array of index locations. An initial search within the chemical mechanism for the phrase "species" is executed, locating the point where the species are listed. A loop is used to iterate forward from this index and extract each species name. This sub-process is contained within a separate function which returns an array of species names.

Within the main script a loop is entered, which iterates over the total species. The first species name is fed as an input to a new function, which returns the CEA coefficients for both viscosity and thermal conductivity. The CEA database text file has consistent spacing and formatting, thus predefined constants are contained within the function that define the character distances between certain locations. For example, the function starts by finding the species index and iterates forward towards the character "V", which marks viscosity. The point at which each temperature boundary value and the 4 coefficients are located remains consistent, seen in the snippet for CH4 below. Using pre-defined constants, the function can skip to these locations and extract each value.

CEA CH4 Coefficients

```
CH4                                V2C2  BOUSHEHRI ET AL (1987)  SVEHLA (1994)
 V  200.0   1000.0   0.57643622E 00-0.93704079E 02 0.86992395E 03 0.17333347E 01
 V 1000.0   5000.0   0.66400044E 00 0.10860843E 02-0.76307841E 04 0.10323984E 01
 C  200.0   1000.0   0.10238177E 01-0.31092375E 03 0.32944309E 05 0.67787437E 00
 C 1000.0   5000.0   0.77485028E 00-0.40089627E 03-0.46551082E 05 0.25671481E 01
```

Two matrices are returned, each containing the coefficients for the specified temperature ranges for viscosity and thermal conductivity respectively. A new function is entered that takes both matrices as inputs and writes a UDF file for each, storing them in the local directory. The MATLAB functions `fopen`, `fprintf` and `fclose` are used to create a new blank file, edit it and save it. The syntax for each file is predefined (see Appendix F) with temperature ranges, coefficients and the file name individually edited per the inputted matrices. On completion the function exited, and the loop in the main script is repeated for all species found in the mechanism. If a species can't be located in the CEA database, no file is generated.

As oxygen is injected as a cryogenic liquid, it requires an additional temperature range that doesn't follow CEA temperature dependant functions, validated only for gaseous and supercritical fluids. Instead, the NIST database was used, with oxygen values for viscosity and thermal conductivity imported into MATLAB alongside the chemical mechanism and CEA database. A 4th order polynomial was fitted to the NIST data up until 170K, slightly beyond the critical point, using the `polyfix` function. This function behaves identically to `polyfit`, with the additional ability to choose exact polynomial endpoints. This enforces a seamless connection with the CEA function, with the starting point specified as the endpoint of the NIST polynomial. The `polyfix` function was sourced externally on the MATLAB file exchange [74].

A2 Mixing Law UDF Generation

The UDF's for each mixing law were manually written in the XCode IDE. For Fluent specific macros and syntax, the customisation manual was used as a reference [48]. A discussion of the viscosity mixing law UDF is presented here as it the most complex example, with differing macros/syntax present in other UDF's described in the customisation manual. Complete mixing law UDF's along with all other UDF's written for this thesis are available in the following appendices.

A requirement for all UDF's is the inclusion of the header file `"udf.h"`, with non-premixed combustion also requiring `"pdf_props.h"`. The `DEFINE` macro is used to set specific parameters within the Fluent simulation. `DEFINE_PROPERTY` is used for fluid properties with 3 arguments required, the UDF name, the cell index (`c`) and the pointer to the cell thread (`t`) on which the property function is to be applied. The cell index and thread are passed to the UDF from the Fluent solver for each individual cell in the mesh, returning a value of datatype `real`. `Real` is a typedef that switches between `float` for single-precision arithmetic, and `double` for double-precision arithmetic, utilised for all variables aside from loop indices. The start of the viscosity mixing law UDF is presented below, capturing the implementation of the aforementioned syntax, macros and datatypes.

```
#include "udf.h"
#include "pdf_props.h"

DEFINE_PROPERTY(wilke_mu,c,t)
{
    real mu_i, mu_j, mf_i, mf_j;
    int i, j;
```

When using the `DEFINE_PROPERTY` macro, auxiliary utilities are available. `Mixture_species_loop` is used to loop through all species in the mixture, requiring a pointer to the mixture, given by `THREAD_MATERIAL`. Additional argument requirements include a species pointer and index integer, defined by the user. `MATERIAL_PROPERTY` returns a pointer to the property array for a given material. `Generic_property` is used to return the `real` value for a specified property, for the given species at the cell temperature. When this command is executed, the species UDF is called and returns the value of the specified property (viscosity or thermal conductivity) based on the temperature `C_T(c,t)`. `PDF_Yi(c,t,j)` is the non-premixed combustion specific function that returns the mass fraction of species `j`. As an example, the snippet below captures the sequential use of these macros, calculating the denominator in Eq.4.53.

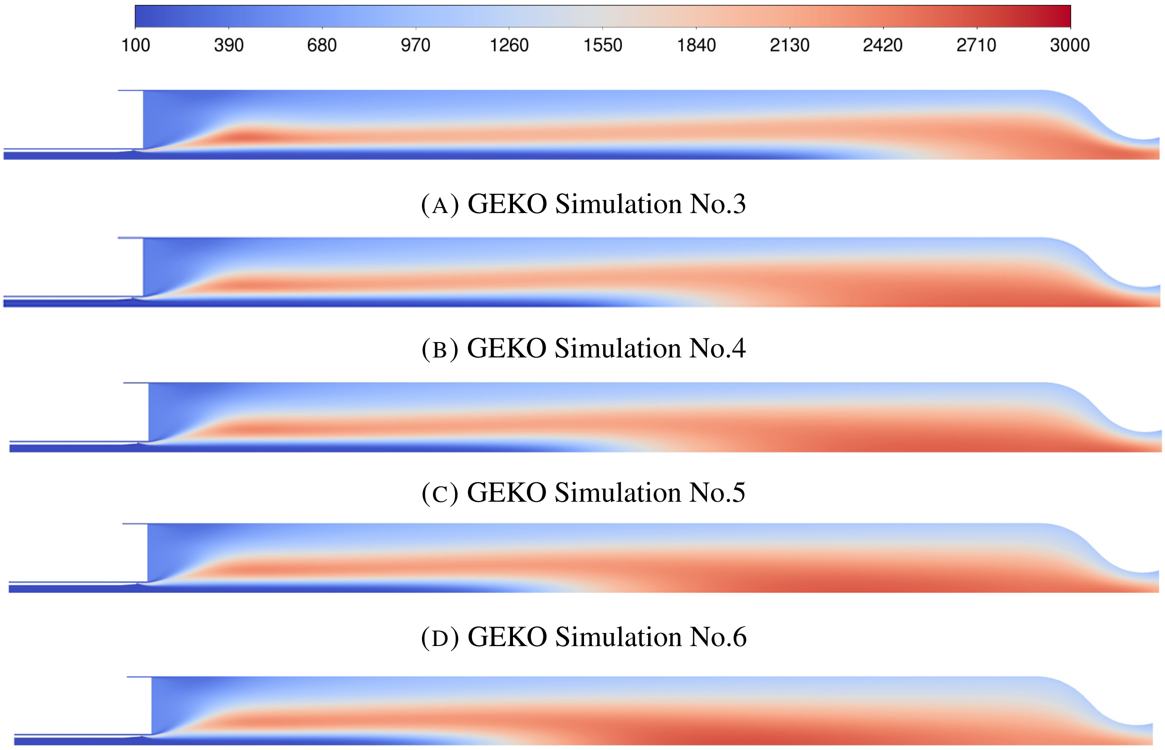
```
mixture_species_loop(THREAD_MATERIAL(t),sp_j,j) {
    prop_j = (MATERIAL_PROPERTY(sp_j));
    mu_j = generic_property(c,t,prop_j,PROP_mu,C_T(c,t));
    mf_j = Pdf_Yi(c, t, j);
    r_u = mu_i/mu_j;
    r_M = mf_i/mf_j;
    r_m = mf_j/mf_i;
    phi = ...
        (1/sqrt(8))*(1/(sqrt(1+r_M)))*pow(1+(sqrt(r_u)*pow(r_m,.25)),2);
    if(isnormal(phi))
        sum_j += phi*mol_i; }
```

Appendix B

GEKO Turbulence Model Contours

Table B.1. GEKO turbulence model coefficients

Simulation No.	CSEP	CNW	CMIX	CJET	CBF_TUR
3	1	1	0.75	0.9	1.5
4	0.75	1	0.75	0.9	1.5
5	0.75	1	1	0.9	1.5
6	0.75	1	1	0.9	1.25
7	0.75	1	1	0.9	1



(A) GEKO Simulation No.3

(B) GEKO Simulation No.4

(C) GEKO Simulation No.5

(D) GEKO Simulation No.6

(E) GEKO Simulation No.7

Figure B.1. Temperature (K) contours of GEKO simulations 3-7 (Tab.B.1)

Appendix C

MATLAB Code

```
%% -----------------------------------------------------------------------------------
%  Main:        flameletAnalyser
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: Uses the readFlameletData function to visualise different
%               parameters, user needs to change constants and files to
%               suit intended purpose
%  Inputs:      Directory - Location of XY data file
%  Outputs:     X_datapoints_ordered - Ordered array of x data points
%               Y_datapoints_ordered - Ordered array of y data points
%% -----------------------------------------------------------------------------------

clear;
clc;

%% Constant Definitions

SSD_text_length = 12;               % Length of Stoichiometric Scalar Dissipation ...
    (SSD) value in chars within .fla file
SSD_expr = 'STOICH_SCADIS';         % String that comes before the SSD value in the ...
    .fla file

temp_text_length = 15;              % Length of temperature value in chars within ...
    .fla file
temp_expr = 'TEMPERATURE';          % String that comes before the temp value in ...
    the .fla file

Z_text_length = 15;                 % Character length of mixture fraction (Z) values
Z_expr = 'Z';                       % Prefix character for mixture fraction (Z) values

```

```
o2_text_length = 15;                % Character length of O2 values
o2_expr = 'massfraction-o2';        % Prefix string for O2 values

%% Read and proces the logarithmic flamelet file

% Read the logarithmic .fla file
log_flamelet_text = fileread('logarithmic_flamelets_BKN19.fla');

% Store temperature in a matrix with each row representing a new SSD value
log_temp_datapoints = readFlameletData(log_flamelet_text,temp_expr,temp_text_length,32);

% Loop through and find the maximum temperature value
log_temp_max = getMax(log_temp_datapoints,32);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
log_SSD_datapoints = readFlameletData(log_flamelet_text,SSD_expr,SSD_text_length,1);

%% Read and process the detailed flamelet file

% Read the detailed .fla file
det_flamelet_text = fileread('32_1.fla');

% Store temperature in a matrix with each row representing a new SSD value
det_temp_datapoints = readFlameletData(det_flamelet_text,temp_expr,temp_text_length,32);

% Loop through and find the maximum temperature value
det_temp_max = getMax(det_temp_datapoints,32);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
det_SSD_datapoints = readFlameletData(det_flamelet_text,SSD_expr,SSD_text_length,1);

%% Read and process the detailed flamelet file with 128 gridpoints

% Read the 128Z .fla file
Z_large_flamelet_text = fileread('Complete_Detailed_Flamelet.fla');

% Store temperature in a matrix with each row representing a new SSD value
Z_large_temp_datapoints = ...
    readFlameletData(Z_large_flamelet_text,temp_expr,temp_text_length,128);

```

```
% Loop through and find the maximum temperature value
Z_large_SSD_datapoints = ...
    readFlameletData(Z_large_flamelet_text,SSD_expr,SSD_text_length,1);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
Z_large_temp_max = getMax(Z_large_temp_datapoints,128);

%% Read and process the RAMEC66 flamelet file with 128 gridpoints

% Read the 128Z .fla file
RAMEC66_flamelet_text = fileread('RAMEC_Flamelet_66.6Bar.fla');

% Store temperature in a matrix with each row representing a new SSD value
RAMEC66_temp_datapoints = ...
    readFlameletData(RAMEC66_flamelet_text,temp_expr,temp_text_length,128);

% Loop through and find the maximum temperature value
RAMEC66_SSD_datapoints = ...
    readFlameletData(RAMEC66_flamelet_text,SSD_expr,SSD_text_length,1);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
RAMEC66_temp_max = getMax(RAMEC66_temp_datapoints,128);

%% Read and process the GRI3.0 flamelet file with 128 gridpoints

% Read the 128Z .fla file
GRI_flamelet_text = fileread('GRI3.0_Flamelet_66.6Bar.fla');

% Store temperature in a matrix with each row representing a new SSD value
GRI_temp_datapoints = ...
    readFlameletData(GRI_flamelet_text,temp_expr,temp_text_length,128);

% Loop through and find the maximum temperature value
GRI_SSD_datapoints = readFlameletData(GRI_flamelet_text,SSD_expr,SSD_text_length,1);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
GRI_temp_max = getMax(GRI_temp_datapoints,128);

%% Read and process the NewM66Bar flamelet file with 128 gridpoints
```

```

% Read the 128Z .fla file
NewM66_flamelet_text = fileread('NewM_Flamelet_66.6Bar.fla');

% Store temperature in a matrix with each row representing a new SSD value
NewM66_temp_datapoints = ...
    readFlameletData(NewM66_flamelet_text,temp_expr,temp_text_length,128);

% Loop through and find the maximum temperature value
NewM66_SSD_datapoints = ...
    readFlameletData(NewM66_flamelet_text,SSD_expr,SSD_text_length,1);

% Store a column vector of SSD values corresponding to the previously
% stored temperature
NewM66_temp_max = getMax(NewM66_temp_datapoints,128);

%% Plot temp vs SSD for both flamelet files

figure(1);

plot(Z_large_SSD_datapoints,Z_large_temp_max,'-o','Color','#e00000','Linewidth',3,...
    'MarkerSize',12,'MarkerEdgeColor','k');
hold on
plot(det_SSD_datapoints,det_temp_max,'-','Color','#d102c0','Linewidth',3,...
    'MarkerSize',12,'MarkerEdgeColor','k');
hold on
plot(log_SSD_datapoints,log_temp_max,'-s','Color','#008fdb','Linewidth',3,...
    'MarkerSize',12,'MarkerEdgeColor','k');
legend('Detailed SSD Distribution | Z - 128','Detailed SSD Distribution | Z - ...
    32','Logarithic SSD Distribution | Z - 32','Fontsize',20);
xlabel('Stoichiometric Scalar Dissipation (s^{-1})','Fontsize',35);
ylabel('Max Temperature (K)','Fontsize',35);
set(gca, 'XScale', 'log')
xlim([5e-4,5e5]);

%% Extract the mixture fraction values

det_Z_datapoints = readFlameletData(det_flamelet_text,Z_expr,Z_text_length,32);
Z_large_Z_datapoints = readFlameletData(Z_large_flamelet_text,Z_expr,Z_text_length,128);

% Refine to a single vector
det_Z_datapoints = det_Z_datapoints(2,:);
```

```
Z_large_Z_datapoints = Z_large_Z_datapoints(2,:);

%% Generate a surface plot containing temperature, Z and SSD

figure(2)
s = [0:0.0078:1];
map = diverging_map(s,[0.230, 0.299, 0.754],[0.706, 0.016, 0.150]);
[X, Y] = meshgrid(Z_large_Z_datapoints,Z_large_SSD_datapoints);
surf(X,Y,Z_large_temp_datapoints);
colormap(map);
title('Surface Plot of Temperature with Z and SSD');
xlabel('Mixture Fraction');
ylabel('Stoichiometric Scalar Dissipation (s^{-1})');
zlabel('Temperature (K)');
set(gca, 'YScale', 'log');


%% Generate 1D scatter plots of Z distribution

figure(3)

subplot(2,1,1);
scatter(Z_large_Z_datapoints,0,'b','LineWidth',1,'SizeData',150);
xlim([0,1]);
set(gca,'ytick',[])
title('128 Z Values');
xlabel('Mixture Fraction')

subplot(2,1,2);
scatter(det_Z_datapoints,0,'b','LineWidth',1,'SizeData',150);
xlim([0,1]);
set(gca,'ytick',[]);
xlabel('Mixture Fraction')
title('32 Z Values');

%% Plot different mechanisms

figure(4);

plot(NewM66_SSD_datapoints,NewM66_temp_max,'-','Color','#008fdb','Linewidth',5,...
    'MarkerSize',12,'MarkerEdgeColor','k');
hold on
```

```
plot(GRI_SSD_datapoints,GRI_temp_max,'-','Color','#d102c0','Linewidth',5,...
    'MarkerSize',12,'MarkerEdgeColor','k');
hold on
plot(RAMEC66_SSD_datapoints,RAMEC66_temp_max,'-','Color','#e00000','Linewidth',5,...
    'MarkerSize',12,'MarkerEdgeColor','k');
hold on
plot(Z_large_SSD_datapoints,Z_large_temp_max,'-','Color',[0,0,1],'Linewidth',5,...
    'MarkerSize',12,'MarkerEdgeColor','k');
legend('NewM','GRI3.0','RAMEC','Zhukov/Kong','Fontsize',55);
xlabel('Stoichiometric Scalar Dissipation (s^{-1})','Fontsize',60);
ylabel('Max Temperature (K)','Fontsize',60);
set(gca, 'XScale', 'log')
xlim([5e-4,7.5e5]);

%% Find maximum value in the given dataset

function max_datapoints = getMax(datapoints,gridpoints)

    for i = 1:length(datapoints(:,1))
        max = 0;
        for j = 1:gridpoints
            if datapoints(i,j) > max
                max = datapoints(i,j);
            end
        end
        max_datapoints(i) = max;
    end

end
```

```
%% -------------------------------------------------------------------------------
%  Function:    readFlameletData
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This function takes a series of flamelet parameter inputs
%               and returns a sorted matrix of specified datapoints over the
%               range of stoichiometric scalar dissipation
%% Inputs
%  Flamelet text - String containing entire .fla file
%  expr - String name of parameter being extracted
%  data_str_length - Length of each parameter value in chars
%  gridpoints - Flamelet table resolution
%
%% Output
%  datapoints - A sorted matrix with each row representing a new
%  stoichiometric scalar dissipation value and each column corresponding to
%  a new grid point
%% ----------------------------------------------------------------------------------
function datapoints = readFlameletData(flamelet_text,expr,data_str_length,gridpoints)

    % Find all index in the flamelet for the expression
    match_index = strfind(flamelet_text,expr);

    % Generate an empty matrix for the datapoints at the correct dimension
    datapoints = zeros(length(match_index),gridpoints);

    % Loop through all the matches
    for i = 1:length(match_index)

        % Initialise an increment variable
        increment = 1;

        % Loop through all the gridpoints
        for j = 1:gridpoints

            % Initialise a flag variable to track the chars
            flag = 0;

            % Loop until an invalid char is recognised
            while flag == 0

                % Initialise the index for the first data point
                data_index = ...
                    (j-1)*(data_str_length+1)+match_index(i)+length(expr)+increment;

                % Check that the ASCII value corresponds to a number
                if flamelet_text(data_index) > 47 && flamelet_text(data_index) < 58

                    % Iterate forward for the length of the value
                    for k = 1:data_str_length

                        % Store each character in a new string
                        data_string(k) = flamelet_text(data_index + k - 1);
                    end

                    % Flag the end of the string has been reached
                    flag = 1;

                % Otherwise increment index forward
                else
                    increment = increment +1;
                end
            end

            % Convert the string to a double and store in the matrix
            data_double = str2double(data_string);
            datapoints(i,j) = data_double;

        end
    end
end
```

```
%% ----------------------------------------------------------------------------------
%  Function:    XY_File_Reader
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This function reads in an XY data file generated by ANSYS
%               Fluent and returns ordered data arrays for plotting purposes
%  Inputs:      Directory - Location of XY data file
%  Outputs:     X_datapoints_ordered - Ordered array of x data points
%               Y_datapoints_ordered - Ordered array of y data points
%% ----------------------------------------------------------------------------------

function [X_datapoints_ordered,Y_datapoints_ordered] = XY_File_Reader(Directory)

    % Read in XY data file as character array
    XY_text = fileread(Directory);

    % Initialise index variables
    X_inc = 1;
    Y_inc = 1;
    file_inc = 1;
```

```
dex = 1;
del = 0;

% Absolute minimum (for data ordering)
min_X = 1e20;
min_Y = 1e20;

% Initialise ordered data arrays
X_datapoints_ordered(1) = 0;
Y_datapoints_ordered(1) = 0;

% Loop through entire XY file
while file_inc ≠ length(XY_text)

    % Check if the current character is a number
    if (XY_text(file_inc) > 47 && XY_text(file_inc) < 58) || (XY_text(file_inc) ...
        == 45 && XY_text(file_inc+1) ≠ 119)

        % Initialise/reset temporary data index
        data_inc = 1;

        % Loop until a carriage return or tab ASCII value
        while XY_text(file_inc+data_inc-1) ≠ 9 && XY_text(file_inc+data_inc-1) ≠...
             13

            % Store each character in a string
            data_string(data_inc) = XY_text(file_inc+data_inc-1);

            % Incremement temporary index
            data_inc = data_inc + 1;
        end

        % Convert the latest string into a double
        data_num = str2double(data_string);

        % Check if the value is an X or Y
        if X_inc == Y_inc

            % Store the new X datapoint
            X_datapoints(X_inc) = data_num;

            % Increment the X index
```

```
                X_inc = X_inc + 1;
            else

                % Store the new Y datapoint
                Y_datapoints(Y_inc) = data_num;

                % Increment the Y index
                Y_inc = Y_inc + 1;
            end

            % Clear the data string ready for a new value
            data_string = data_string([]);

            % Increment the file index by the temporary data index
            file_inc = file_inc + data_inc;

        else

            % Increment to the next character if a number is not found
            file_inc = file_inc + 1;
        end
    end

    % Retrieve the amount of datapoints
    len = length(X_datapoints);

    % In the case the data isn't ordered out of Fluent, reorder
    while length(X_datapoints_ordered) < len
        for i = 1:length(X_datapoints)

            % Check if a new minimum is found
            if X_datapoints(i) < min_X || isnan(X_datapoints(i))
                min_X = X_datapoints(i);
                min_Y = Y_datapoints(i);
                del = i;
            end
        end

        % Delete the new minimum from the unordered arrays
        X_datapoints(del) = [];
        Y_datapoints(del) = [];

        % Place the new minimum in the ordered array
        X_datapoints_ordered(dex) = min_X;
        Y_datapoints_ordered(dex) = min_Y;

        % Increment loop index and redeinfe the absolute minimum
        dex = dex+1;
        min_X = 1e20;
        min_Y = 1e20;
    end
end
```

```
%% ----------------------------------------------------------------------------------
%  Main:        transport_reader
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This algorithm reads in a specifed chemical mechansim and the CEA ...
    database
%               and generates a series of UDF's for each species, defining ...
    viscosity and
%               thermal conductivity as function of temperature
%% ----------------------------------------------------------------------------------

clear;
clc;

% Read in chemical mechanism
chem_mech = fileread('NewMechanism.txt');

% Read in CEA transport coefficients
transport_data = fileread('trans.txt');

% Read in Oxygen NIST data
O2_NIST = readmatrix('O2_NIST.txt');

% Extract NIST temperature, viscosity (mu) and thermal conductivity (ktc)
NIST_temp = O2_NIST(:,1);
NIST_mu = O2_NIST(:,12);
NIST_ktc = O2_NIST(:,13);
idx = 1;

```

```
% Remove data from the NIST arrays above 170 K
while NIST_temp(idx) < 170
    idx = idx + 1;
end

NIST_temp = NIST_temp(1:idx);
NIST_mu = NIST_mu(1:idx);
NIST_ktc = NIST_ktc(1:idx);

% Extract the list of species from the chemical mechanism
species_list = speciesFind(chem_mech);

% Loop through all species and generate a UDF for both thermal conductivity
% and viscosity
for UDF = 1:length(species_list)

    % Extract two copies of the current species name
    species_trans = species_list(UDF);
    species_name = species_list(UDF);

    % Define the amount of characters in the species name
    len = strlength(species_trans);

    % Define key markers within the CEA transport database
    visc = 'V ';
    cond  = 'C ';
    V_dist = 60;
    C_dist = 306;
    space = 82;
    whitespace = 34;

    % Loop through and append the correct amount of spaces that corresponds
    % to the amount whitespace present in the CEA transport file for current species
    for i = 1:(whitespace-len)
        species_trans = strcat(species_trans, {' '});
    end

    % Find the location of the current species in the CEA database
    species_index = strfind(transport_data,species_trans);

    % Check the species index is not 0
    if species_index > 0
```

```

        % Define two indices, one for viscosity and one for thermal conductivity
        dx_v = species_index+1;
        dx_c = species_index+1;

        % Find the location of the viscosity coefficients
        while transport_data(dx_v) ≠ 'V' || transport_data(dx_v+1) ≠ ' '
            dx_v = dx_v+1;
        end

        % Find the location of the thermal conductivity coefficients
        while transport_data(dx_c) ≠ 'C' || transport_data(dx_c+1) ≠ ' '
            dx_c = dx_c+1;
        end

        % Extract the temperature breaks and coefficients for the current species
        [visc_breaks, visc_coef_A, visc_coef_B, visc_coef_C, visc_coef_D] = ...
            extractCoeff('V',space,dx_v,transport_data);
        [cond_breaks, cond_coef_A, cond_coef_B, cond_coef_C, cond_coef_D] = ...
            extractCoeff('C',space,dx_c,transport_data);

        % Transpose each coefficient vector and store in matrix
        visc_coeff = [visc_coef_A',visc_coef_B',visc_coef_C',visc_coef_D'];
        cond_coeff = [cond_coef_A',cond_coef_B',cond_coef_C',cond_coef_D'];

        % Write the viscosity and thermal conductivity UDF for each species
        writeTransportUDF(species_name,visc_breaks,visc_coeff,cond_breaks,cond_coeff);

    end
end

%% ----------------------------------------------------------------------------------
%  Function:    extractCoeff
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This function locates and returns the CEA coefficients and temperature
%               boundaries for a given species
%% ----------------------------------------------------------------------------------

function [breaks, coef_A, coef_B, coef_C, coef_D] = ...
    extractCoeff(trans_marker,space,dx,transport_data)
    for i = 0:2
```

```
        if transport_data(dx+i*(space)) == trans_marker

            for j = 1:7
                str(j) = transport_data(dx+(i*space)+j);
            end
            num = str2double(str);

            breaks(i+1) = num;
            coef_A(i+1) = strConvert(20,33,transport_data,i,dx,space);
            coef_B(i+1) = strConvert(34,48,transport_data,i,dx,space);
            coef_C(i+1) = strConvert(49,63,transport_data,i,dx,space);
            coef_D(i+1) = strConvert(64,78,transport_data,i,dx,space);

        else
            break
        end
    end
end


function num = strConvert(lower_bound,upper_bound,data,i,dx,space)

    for j = lower_bound:upper_bound
        if data(dx+(i*space)+j) == ' '
            str_A(j-(lower_bound-1)) = '+';
        else
            str_A(j-(lower_bound-1)) = data(dx+(i*space)+j);
        end
    end
    num = str2double(str_A);
end

%% ----------------------------------------------------------------------------------
%  Function:    speciesFind
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This function locates the list of species in the inputted chemical
%               mechanism and returns them as an array of strings
%% ----------------------------------------------------------------------------------
function species_list = speciesFind(chem_mech)

species_s = "SPECIES";
```

```

mech_species_index = strfind(chem_mech,species_s)+7;
species_list = strings;
sdx = 1;
sldx = 1;

while chem_mech(mech_species_index) ≠ 'E' || chem_mech(mech_species_index+1) ≠ 'N'

    if (chem_mech(mech_species_index) ≥ 40 && chem_mech(mech_species_index) ≤ 57) ...
        || (chem_mech(mech_species_index) ≥ 65 && chem_mech(mech_species_index) ≤ 90)

        while (chem_mech(mech_species_index) ≥ 40 && chem_mech(mech_species_index) ≤...
             57) || (chem_mech(mech_species_index) ≥ 65 && ...
            chem_mech(mech_species_index) ≤ 90)
            s_temp(sdx) = chem_mech(mech_species_index);
            if s_temp(sdx) == 'R'
                s_temp(sdx) = 'r';
            end
            if s_temp(sdx) == 'E'
                s_temp(sdx) = 'e';
            end
            if s_temp(sdx) == 'I'
                s_temp(sdx) = 'i';
            end
            sdx = sdx+1;
            mech_species_index = mech_species_index +1;
        end

        species_list(sldx) = [s_temp];
        s_temp = '';
        sldx = sldx + 1;
        sdx = 1;
    end
    mech_species_index = mech_species_index +1;
end

end

%% --------------------------------------------------------------------------------
%  Function:    writeTransportUDF
%  Author:      Ben McNutt
%  Project:     ESIPS DLR Thesis 2022
%  Description: This function writes a species viscosity and thermal
%               conductivity UDF in the required ANSYS Fluent format
%% ----------------------------------------------------------------------------------
function writeTransportUDF(species_name, visc_breaks, visc_coeff, cond_breaks, ...
    cond_coeff)

        % Viscosity
        file_name = strcat(species_name,'_visc.c');
        fileID = fopen(file_name,'w');

        fprintf(fileID,'#include "udf.h"\nDEFINE_PROPERTY(%s_viscosity,c,t)\n{\n  ...
            real mu_%s;\n  real temp = C_T(c,t);\n', species_name, species_name);
        fprintf(fileID,'  if (temp < %.2f)\n', visc_breaks(2));
        fprintf(fileID,'    mu_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
            (%.8f/pow(temp,2)) + %.8f)/10000000;\n', species_name, visc_coeff(1,1), ...
            visc_coeff(1,2), visc_coeff(1,3), visc_coeff(1,4));

        if length(visc_breaks) == 3
            fprintf(fileID,'  else if (temp ≥ %.2f && temp < %.2f)\n', ...
                visc_breaks(2), visc_breaks(3));
            fprintf(fileID,'    mu_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000000;\n', species_name, ...
                visc_coeff(2,1), visc_coeff(2,2), visc_coeff(2,3), visc_coeff(2,4));
            fprintf(fileID,'  else\n');
            fprintf(fileID,'    mu_%s = (%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000000;\n', species_name, ...
                visc_coeff(3,1), visc_coeff(3,2), visc_coeff(3,3), visc_coeff(3,4));
        else
            fprintf(fileID,'  else\n');
            fprintf(fileID,'    mu_%s = (%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000000;\n', species_name, ...
                visc_coeff(2,1), visc_coeff(2,2), visc_coeff(2,3), visc_coeff(2,4));
        end

        fprintf(fileID,'  return mu_%s;\n}\n', species_name);
        fclose(fileID);

        % Thermal conductivity
        file_name = strcat(species_name,'_ktc.c');
        fileID = fopen(file_name,'w');

        fprintf(fileID,'#include "udf.h"\nDEFINE_PROPERTY(%s_ktc,c,t)\n{\n  real ...
            ktc_%s;\n  real temp = C_T(c,t);\n', species_name, species_name);
        fprintf(fileID,'  if (temp < %.2f)\n', cond_breaks(2));
        fprintf(fileID,'    ktc_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
            (%.8f/pow(temp,2)) + %.8f)/10000;\n', species_name, cond_coeff(1,1), ...
            cond_coeff(1,2), cond_coeff(1,3), cond_coeff(1,4));

        if length(visc_breaks) == 3
            fprintf(fileID,'  else if (temp ≥ %.2f && temp < %.2f)\n', ...
                cond_breaks(2), cond_breaks(3));
            fprintf(fileID,'    ktc_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000;\n', species_name, ...
                cond_coeff(2,1), cond_coeff(2,2), cond_coeff(2,3), cond_coeff(2,4));
            fprintf(fileID,'  else\n');
            fprintf(fileID,'    ktc_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000;\n', species_name, ...
                cond_coeff(3,1), cond_coeff(3,2), cond_coeff(3,3), cond_coeff(3,4));
        else
            fprintf(fileID,'  else\n');
            fprintf(fileID,'    ktc_%s = exp(%.8f*log(temp) + (%.8f/temp) + ...
                (%.8f/pow(temp,2)) + %.8f)/10000;\n', species_name, ...
                cond_coeff(2,1), cond_coeff(2,2), cond_coeff(2,3), cond_coeff(2,4));
        end

        fprintf(fileID,'  return ktc_%s;\n}\n', species_name);
        fclose(fileID);
end
```

APPENDIX D

Embedded Wall Temperature Profile UDF

```
#include "udf.h"
DEFINE_PROFILE(Thin_Wall_Profile,t,i)
{
  real x[ND_ND];
  face_t f;
  begin_f_loop(f,t)
  {
    F_CENTROID(x,f,t);
    if(x[0] < 0.068)
    {
         F_PROFILE(f,t,i) = 265;
    }
    else if(x[0] ≥ 0.068 && x[0] < 0.1435)
    {
         F_PROFILE(f,t,i) = 427325.20000*pow(x[0],3) + -180209.87991*pow(x[0],2) + ...
             25876.49684*x[0] + -792.67823;
    }
    else
    {
         F_PROFILE(f,t,i) = 517.59432*x[0] + 398.138;
    }
  }
  end_f_loop(f,t)
}
```

APPENDIX E

Transport Mixing Law UDF's

```
#include "udf.h"
#include "pdf_props.h"

DEFINE_PROPERTY(wilke_mu,c,t)
{
    int i, j;
    real rat_mu, rat_M, rat_m;
    Material *sp_i, *sp_j;
    real mu_i, mu_j;
    real mf_i, mf_j;
    real rho_i, rho_j;
    real mol_i, mol_j;
    real mw_i, mw_j;
    real phi;
    Property *prop_i, *prop_j;
    real sum_i = 0.;
    real sum_j = 0.;
    mixture_species_loop(THREAD_MATERIAL(t),sp_i,i)
    {
        sum_j = 0.;
        prop_i = (MATERIAL_PROPERTY(sp_i));
        mu_i = generic_property(c,t,prop_i,PROP_mu,C_T(c,t));
        mf_i = Pdf_Yi(c, t, i);
        rho_i = C_R(c,t);
        mw_i = MATERIAL_PROP(sp_i,PROP_mwi);
        mol_i = 1000*mf_i*rho_i/mw_i;
        if(isnormal(mol_i) && isnormal(mu_i))
        {
            mixture_species_loop(THREAD_MATERIAL(t),sp_j,j)
            {
                prop_j = (MATERIAL_PROPERTY(sp_j));
                mu_j = generic_property(c,t,prop_j,PROP_mu,C_T(c,t));
                mf_j = Pdf_Yi(c, t, j);
                rat_mu = mu_i/mu_j;
                rat_M = mf_i/mf_j;
                rat_m = mf_j/mf_i;
                phi = ...
                    (1/sqrt(8))*(1/(sqrt(1+rat_M)))*pow(1+(sqrt(rat_mu)*pow(rat_m,0.25)),2);
                if(isnormal(phi))
                    sum_j += phi*mol_i;
            }
            if isnormal(sum_j)
                sum_i += (mol_i*mu_i)/sum_j;
        }
    }
    return sum_i;
}
```

```
#include "udf.h"
#include "pdf_props.h"

DEFINE_PROPERTY(zipp_ktc,c,t)
{
    int i, int j;
    Material *sp_i, *sp_j;
    real ktc;
    real mf_i, mf_j;
    real rat_m;
    real rho_i, rho_j;
    real mol_i, mol_j;
    real mw_i, mw_j;
    real phi;
    Property *prop_i;
    Property *prop_j;
    real sum = 0.;
    mixture_species_loop(THREAD_MATERIAL(t),sp_i,i)
    {
        phi = 0.;
        prop_i = (MATERIAL_PROPERTY(sp_i));
        ktc = generic_property(c,t,prop_i,PROP_ktc,C_T(c,t));
        mf_i = Pdf_Yi(c, t, i);
        rho_i = C_R(c,t);
        mw_i = MATERIAL_PROP(sp_i,PROP_mwi);
        mol_i = 1000*mf_i*rho_i/mw_i;
        if(isnormal(mol_i) && isnormal(ktc))
        {
            mixture_species_loop(THREAD_MATERIAL(t),sp_j,j)
            {

                prop_j = (MATERIAL_PROPERTY(sp_j));
                mf_j = Pdf_Yi(c, t, j);
                rho_j = C_R(c,t);
                mw_j = MATERIAL_PROP(sp_i,PROP_mwi);
                mol_j = 1000*mf_j*rho_j/mw_j;
                rat_m = mf_j/mf_i;
                if(isnormal(rat_m)&&isnormal(mol_j))
                    phi += mol_j*sqrt(rat_m);
            }
            if isnormal(phi)
                sum += (mol_i*ktc)/phi;
        }
    }
    return sum;
}
```

APPENDIX F

CEA Species Transport UDF's

The viscosity and thermal conductivity UDF's for O_2 and CH_3OH are included below. All UDF's follow the same structure and can be generated using the transportReader MATLAB function supplied in Appendix C. O_2 is the exception with an additional temperature range.

```
#include "udf.h"
DEFINE_PROPERTY(O2_ktc,c,t)
{
  real ktc_O2;
  real temp = C_T(c,t);
  if (temp < 169)
    ktc_O2 = 0.00000000120710043919*pow(temp,4) -0.00000054669353726675*pow(temp,3) ...
        + 0.00008696006216142521*pow(temp,2) -0.00710599791053700188*temp + ...
        0.41081319636404117768;
  else if (temp ≥ 169 && temp < 1000.00)
    ktc_O2 = exp(0.77229167*log(temp) + (6.84632100/temp) + ...
        (-5893.33770000/pow(temp,2)) + 1.22103650)/10000;
  else if (temp ≥ 1000.00 && temp < 5000.00)
    ktc_O2 = exp(0.90917351*log(temp) + (291.24182000/temp) + ...
        (-79650.17100000/pow(temp,2)) + 0.06485163)/10000;
  else
    ktc_O2 = exp(1.12182620*log(temp) + (-19286.37800000/temp) + ...
        (23295011.00000000/pow(temp,2)) + 20.34204300)/10000;
  return ktc_O2;
}
```

```
#include "udf.h"
DEFINE_PROPERTY(O2_viscosity,c,t)
{
  real mu_O2;
  real temp = C_T(c,t);
  if (temp < 169)
    mu_O2 = (0.00001431539893805554*pow(temp,4) -0.00805529654977796793*pow(temp,3) ...
        + 1.67082505295016292557*pow(temp,2) -153.05558231455950135569*temp + ...
        5366.27486337179652764462)/1000000;
  else if (temp ≥ 169 && temp < 1000.00)
    mu_O2 = exp(0.60916180*log(temp) + (-52.24484700/temp) + ...
        (-599.74009000/pow(temp,2)) + 2.04108010)/10000000;
  else if (temp ≥ 1000.00 && temp < 5000.00)
    mu_O2 = exp(0.72216486*log(temp) + (175.50839000/temp) + ...
        (-57974.81600000/pow(temp,2)) + 1.09010440)/10000000;
  else
    mu_O2 = (0.73981127*log(temp) + (391.94906000/temp) + ...
        (-378331.68000000/pow(temp,2)) + 0.90931780)/10000000;
  return mu_O2;
}
```

```
#include "udf.h"
DEFINE_PROPERTY(CH3OH_ktc,c,t)
{
  real ktc_CH3OH;
  real temp = C_T(c,t);
  if (temp < 1000.00)
    ktc_CH3OH = exp(0.33374512*log(temp) + (-1161.71540000/temp) + ...
        (108942.11000000/pow(temp,2)) + 5.76841240)/10000;
  else if (temp ≥ 1000.00 && temp < 5000.00)
    ktc_CH3OH = exp(0.42733576*log(temp) + (-1268.25280000/temp) + ...
        (209004.63000000/pow(temp,2)) + 5.12838600)/10000;
  else
    ktc_CH3OH = exp(1.05183580*log(temp) + (-4255.59440000/temp) + ...
        (14288688.00000000/pow(temp,2)) + -0.88950473)/10000;
  return ktc_CH3OH;
}
```

```
#include "udf.h"
DEFINE_PROPERTY(CH3OH_viscosity,c,t)
{
  real mu_CH3OH;
  real temp = C_T(c,t);
  if (temp < 1000.00)
    mu_CH3OH = exp(0.58408390*log(temp) + (-306.77174000/temp) + ...
        (27569.89200000/pow(temp,2)) + 1.97943480)/10000000;
  else if (temp ≥ 1000.00 && temp < 5000.00)
    mu_CH3OH = exp(0.61454903*log(temp) + (-165.40203000/temp) + ...
        (-27881.99500000/pow(temp,2)) + 1.68307130)/10000000;
  else
    mu_CH3OH = (0.72150912*log(temp) + (750.12895000/temp) + ...
        (-1182550.70000000/pow(temp,2)) + 0.85493645)/10000000;
  return mu_CH3OH;
}
```