DEVELOPMENT OF A STATIC SHAFT WANKEL EXPANDER FOR SMALL-SCALE APPLICATIONS

By

Gavin Tozer

A thesis submitted to the University of Birmingham for the degree of

DOCTOR OF PHILOSOPHY

Mechanical Engineering

School of Engineering

College of Engineering and Physical Sciences

University of Birmingham

June 2020

UNIVERSITY OF
BIRMINGHAM

University of Birmingham Research Archive

e-theses repository

This unpublished thesis/dissertation is copyright of the author and/or third parties. The intellectual property rights of the author or third parties in respect of this work are as defined by The Copyright Designs and Patents Act 1988 or as modified by any successor legislation.

Any use made of information contained in this thesis/dissertation must be in accordance with that legislation and must be properly acknowledged. Further distribution or reproduction in any format is prohibited without the permission of the copyright holder.

ABSTRACT

With increasing global demand for energy and the problems of climate change from extensive use of fossil fuels new ways to increase the renewable energy sources and reduce energy waste are required. Energy storage, such as liquid air energy storage (LAES), is one way to improve renewable energy utilisation. The organic Rankine cycle (ORC) is a system that allows the recovery of low-grade heat energy, which can be in the form of waste heat or geothermal heat. Both of these technologies are also good for distributed energy production, which reduces losses associated with energy transport.

This thesis looks at the development of a Wankel gas expansion device for use in gas liquefaction or ORC systems. It starts with an extensive literature review into LAES and ORC systems. In which literature shows a clear need for the development of both gas liquefaction systems and small-scale, low-cost and efficient gas expanders. A review of available gas expanders is then presented followed by a detailed review of Wankel expansion devices. This review concludes that the Wankel expander has many qualities making it suitable for small-scale low-cost systems but has the issues of friction and the requirement for external valves which need to be addressed.

The next part of the thesis describes the creation of numerical models to simulate gas liquefaction systems. The results of these models suggest that the Kapitza system performs the best, and the most effective way to increase its efficiency is improving the performance of the expansion device. Next, the

creation of computational fluid dynamics (CFD) models for two different types of Wankel expander are described. One type is a standard Wankel expander with side ports, whilst the second is a newly designed static shaft Wankel expander based on the original DKM Wankel engine. The results of the CFD simulations show the main drawback of the standard Wankel expander is that it only reaches a maximum isentropic efficiency of 64.88%, due to lack of inlet control. The static shaft Wankel expander simulations show that it could reach isentropic efficiencies of 87.35% and can be designed for a large range of inlet pressures. CFD is also performed using two organic working fluids often used in organic Rankine cycle (ORC) systems, giving maximum isentropic efficiency of 85.6%. This demonstrates that the expander could be used in an ORC system and achieve similar performance to compressed air systems.

Finally, an experimental test rig was designed and two static shaft Wankel expander prototypes were manufactured, one from plastic and one from metal. The prototypes were tested to find their power output and isentropic efficiency performances. Problems with the initial prototype design were found and a second prototype design was manufactured to address these issues. The experimental results were found to agree well with the CFD power output (average of 5.4W deviation), but with a much higher deviation for the CFD efficiency (average of 10.1% deviation), this was thought to be due to leakages not accounted for in the CFD models.

ACKNOWLEDGMENTS

For all the invaluable guidance, ideas, encouragement and many hours of proof-reading papers and thesis chapters, I present my eternal gratitude to my supervisors, **Dr Raya Al-Dadah** and **Dr Saad Mahmoud**. They were also responsible for encouraging me to undertake this PhD when I was a master's student under them, therefore I would not have even started it without their belief in me.

For the invaluable help and advice with all the experimental work, I would like to thank **Simon Rowan**. For the extensive advice on the prototypes' designs for manufacturing and for paying so much care and attention in machining them, I thank **Steve Brookes** and **Warren Hay**. I also thank all my friends and colleagues at the University of Birmingham and beyond, for all the help and support throughout the years.

I would like to thank my parents **Erica Tozer** and **Garth Tozer** for always supporting me in all aspects of my life, making me the person I am today. Last, but definitely not least, I give all my thanks to the wonderful **Stephanie Brett**, for all the love and support during the final stages of my PhD.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Symbols

A	Area of the rear of an apex seal [m^2]
e	Wankel eccentricity [m]
E	Heat exchanger effectiveness
f	Force [N]
h	Specific enthalpy [J/kg]
I	Current [A]
m	Mass [kg]
$\dot{m}$	Mass flow rate [kg/s]
N	Number of sample points
p	Pressure [Pa]
P	Power output [W]
R	Wankel generating radius [m]
r_{COM}	Radius of circular path the apex seals' centre of masses travel [m]
$r_{contact}$	Mean radius the apex seals' contact point with housing travels [m]
s	Specific entropy [J/kg.K]
T	Temperature [K]
t	Time [s]
U	Sensor uncertainty
v	Velocity [m/s]
V	Voltage [V]
$\dot{V}$	Volume flow rate [m^3/s]
w	Ideal specific work to liquefy a gas [J/kg]

W_{pv}	Work calculated from area under pressure-volume diagram [J]
X	Fraction diverted to expander(s)
x_h	Cartesian x coordinate of generated housing shape
y_h	Cartesian y coordinate of generated housing shape
η	Isentropic efficiency [%]
θ	Housing generating angle
μ	Coefficient of friction
ρ	Density [kg/m^3]
τ	Torque output [Nm]
$\bar{\bar{\tau}}$	Stress tensor
ω	Rotational speed [rev/s]

Subscripts

1	At ambient conditions
actual	The actual value
centripetal	Centripetal force
CFD	CFD output value
compressor	Relating to a compressor
expander	Relating to an expander
f	At saturated liquid conditions
friction	Relating to apex seal friction
i	At iteration i
ideal	The ideal theoretical value
inlet	At inlet to device
outlet	At the outlet of the device

pressure	Pressure force
random	From calculated random errors
seal	Relating to an apex seal
systematic	From calculated systematic errors

Abbreviations

CAD	Computer aided design
CES	Cryogenic energy storage
CFD	Computational fluid dynamics
CNC	Computer numerical control
DKM	Drehkolbenmotor (Purely rotary type Wankel expander)
FEA	Finite element analysis
FOM	Figure of merit
ICE	Internal combustion engine
KKM	Kreiskolbenmotor (Planetary motion type Wankel engine)
LAES	Liquid air energy storage
LNG	Liquefied natural gas
ORC	Organic Rankine cycle
PIV	Particle image velocimetry
PWM	Pulse width modulated
UDF	User defined function

PUBLICATIONS

- Sadiq, G.A., Tozer, G.M., Mahmoud, S.M. and Al-Dadah, R.K., *Numerical investigation of the two stage Wankel expander performance*. International Conference for Students on Applied Engineering (ICSAE) in Newcastle upon Tyne, 2016. p. 345-350

- Sadiq, G.A., Tozer, G., Al-Dadah, R., Mahmoud, S., *CFD simulations of compressed air two stage rotary Wankel expander–Parametric analysis*. Energy conversion and management, 2017. **142**: p. 42-52.

- Tozer, G., Al-Dadah, R., and Mahmoud, S., *Effect of the apex gap size on the performance of a small scale Wankel expander*. in HEAT POWERED CYCLES, 2018. University of Bayreuth, Germany.

- Tozer, G., Al-Dadah, R., Mahmoud, S., *Simulating apex gap sizes in a small scale Wankel expander for air liquefaction*. Applied Thermal Engineering. 2019 May 25;154:476-84.

- Tozer, G., Ga, J.L., Al-Dadah, R., and Mahmoud, S., *Development of efficient static shaft Wankel expander for organic Rankine cycles*, 5th International Seminar on ORC Power Systems, 2019. Athens, Greece.

CHAPTER 1

INTRODUCTION

The world has an increasing demand for energy and special attention has been given to renewable energy that can satisfy this need with less carbon emissions, this can be seen in the increase in wind and solar energy supply from 36,560 ktoe in 1990 to 256,830 ktoe in 2017 [1]. One of the biggest disadvantages of most renewable energy sources is their intermittent nature [2], data from International Energy Agency [1] shows that in 2017 only 17.63% of wind and solar energy supplied was consumed. Therefore, affordable, efficient and geographically viable energy storage methods are being researched to help reduce this problem by storing the energy available at off-peak times. One promising energy storage technology is liquid air energy storage (LAES) [2], which has a predicted round-trip efficiency of up to 55% [3] for a stand-alone system or up to 84.15% [4] if part of a hybrid system.

Another method to increase the world's available energy is to reduce energy lost to heat and exploit wasted heat energy sources. These include heat wasted in industrial processes, engines, steam powerplants and geothermal sources. As the amount of wasted heat is approximately 50% of the world energy consumption, there is a massive improvement opportunity. The wasted heat could be transformed into usable energy using a technology such as the organic Rankine cycle (ORC) which has recently received a surge in

development [5] and is capable of producing energy from heat sources with temperatures in the low-range of 60-300°C [5]. ORC systems also provide the opportunity for more distributed energy production, this would reduce the energy wasted in transporting it through power lines, which is generally 12% of the power [5].

LAES and ORC systems both make use of gas expansion devices and for both it is one of the most critical components affecting performance. The expansion devices are also commonly the most expensive component in both systems. Therefore, development of a low-cost, reliable and efficient gas expansion device is desirable. The Wankel engine was known for its low cost, reliable nature with a high power to weight ratio. However, there is little research on Wankel gas expansion devices [5, 6]. This thesis follows research and development of a low-cost, more efficient Wankel expander.

1.1. Aim and Objectives

Thesis Aim:

This research aims to develop a simple, low-cost, small-scale Wankel expansion device for use within an LAES or ORC system to help encourage the development of these technologies by making them more affordable whilst maintaining performance.

Thesis Objectives:

1. Review research on the state of LAES and ORC development, on expansion devices available and their attributes and on the

developments of Wankel technologies (including Wankel expanders) to ascertain if the Wankel expander has the potential as a solution.

2. Develop thermodynamic numerical models for the gas liquefaction processes of LAES and use these models to find how the expander performance affects the cycle performance.
3. Create an initial design for a Wankel expander and develop computational fluid dynamics (CFD) models to predict its performance at various operating conditions, different working fluids (air and organic) and design modifications.
4. Design an experimental test rig to measure the performance of a Wankel expansion device running on compressed air.
5. Design and manufacture a prototype Wankel expander to assess the practicalities and test its performance to validate the CFD models.

1.2. Thesis outline

CHAPTER ONE: An introduction to the energy problems and the methods and solutions that will be developed.

CHAPTER TWO: A comprehensive literature review on research involving Wankel expansion devices and similar technology. Also included is a literature review on the state of energy in the world, LAES, ORC, gas liquefaction cycles, and gas expansion devices.

CHAPTER THREE: An introduction to the newly designed static shaft Wankel expander followed by a description of the developed CFD models for

both standard Wankel expanders and static shaft Wankel expanders, with and without apex seals. Finally, the results of the CFD simulations run with various operating conditions and design parameters are presented and discussed.

CHAPTER FOUR: This chapter describes the design of an experimental test rig to measure the performance of small-scale Wankel expansion devices. The sensors used and their calibration are described and the test procedure is laid out.

CHAPTER FIVE: The design of the two prototype expanders used with the test rig are described. Their measured performances are presented and discussed as are problems with the test procedure.

CHAPTER SIX: Conclusions to the research conducted within this thesis are presented and recommendations for future work are given.

CHAPTER 2

LITERATURE REVIEW

2.1. Chapter Introduction

This chapter looks at the current and past literature relating to the work in this thesis. It starts with an analysis on the energy usage and supply in the world and how the work proposed in the thesis may help address the rising issues. Following this, a review on Liquid Air Energy Storage (LAES) and associated gas liquefaction processes is presented. The Organic Rankine Cycle is also introduced and critically reviewed on its relevance to this work. Finally, an in-depth study of different gas expansion devices and then specifically the Wankel expansion devices is shown.

2.2. Global State of Energy

Global energy demand is increasing in all sectors, but particularly in industry and transport, as seen in Figure 2.1 [1]. Figure 2.2 shows the global energy consumption by source, showing that consumption of energy from the wind and solar sources are increasing, even though they still make up only a tiny fraction of the overall consumption [1]. Figure 2.3 shows the global energy supply by its source, the fraction of wind and solar *supply* is much higher than *consumption [1]*. Therefore, a lot of the wind and solar energy supply is not

currently utilised, this is probably due to the intermittent nature of these sources as they often supply energy at times of low demand.

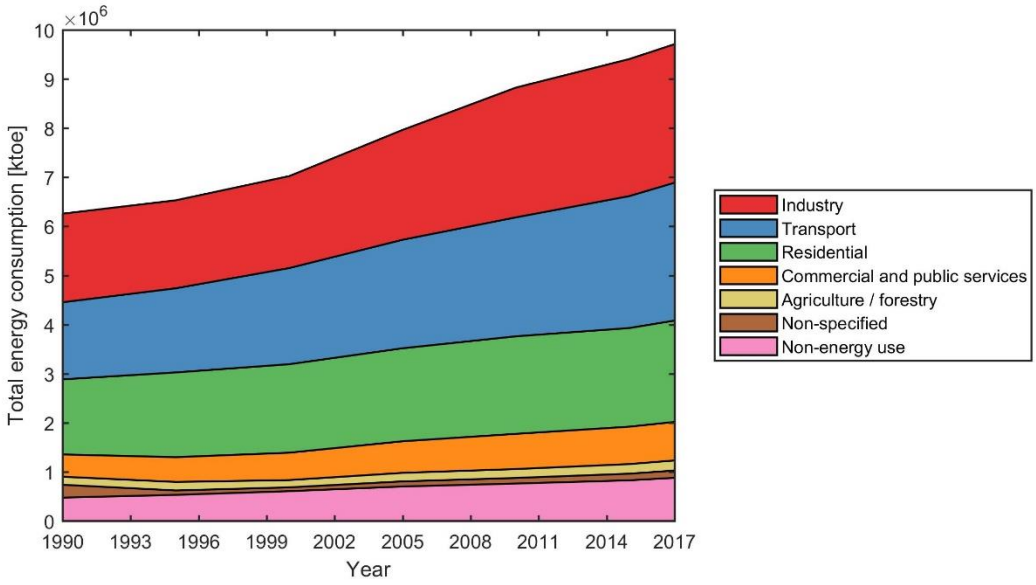


Figure 2.1 – Global energy consumption by sector (data from International Energy Agency [1])

To reduce this wasted energy, it would be extremely beneficial to be able to store this large amount of wasted off-peak energy supply, for use at peak times. There are many different methods for energy storage that have their own advantages and disadvantages. One of the most common forms of energy storage is via lithium ion batteries. Although these by far have the best storage efficiency (up to 95%) and energy density [7], they are also very expensive, requiring rare minerals and have a short lifetime [2]. These two factors make them not viable for storing large amounts of energy.

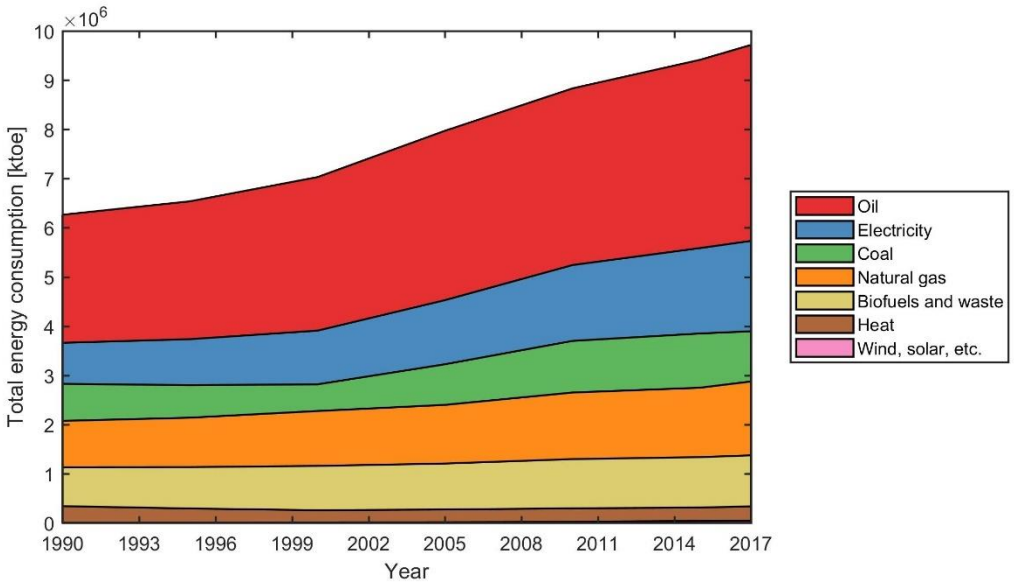


Figure 2.2 – Global energy consumption by source (data from International Energy Agency [1])

The most common energy storage method for large quantities of energy is pumped hydro energy storage. In this method water is pumped up stream into large reservoirs, where it is stored until the energy is needed, at which time the water can be allowed to flow back down stream powering turbines. This method is generally less efficient than batteries due to mechanical and pipe flow losses resulting in an energy efficiency of 70-80% [8]. However, it allows the storage of vast quantities of electricity and has a very low cost to set up. Unfortunately, it also requires very specific geographical settings, where there is a large enough space for a reservoir and a steep enough gradient in the terrain [2].

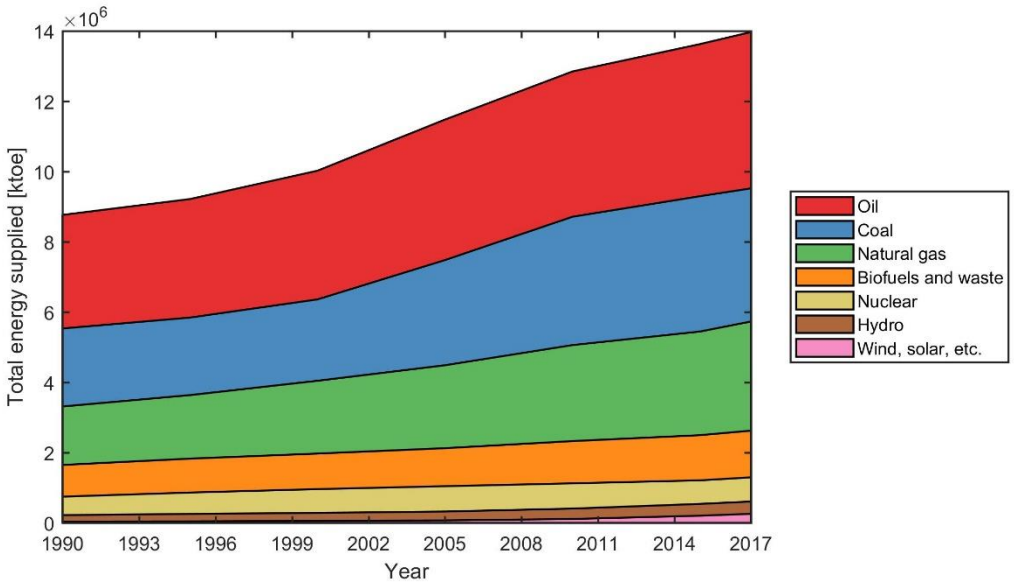


Figure 2.3 – Global energy supply by source (data from International Energy Agency [1])

Compressed air energy storage is another energy storage method, in which air is pumped into underground caverns, where it is stored at high pressures and can later be released to drive turbines. Like pumped hydro, this method allows relatively large amounts of energy to be stored at a low setup cost. Furthermore, it also requires specific geographical requirements. However, this method requires the presence of underground caverns, in the form of depleted aquifers or natural caverns. Therefore, both these methods will soon run short of available locations [2]. Two relatively new energy storage technologies are thermochemical energy storage and liquid air energy storage (LAES). Thermochemical energy storage utilises endothermic and exothermic chemical reactions to store and release energy, with a maximum round-trip efficiency of 44% being reported [9]. LAES provides a large capacity of energy storage and has much less geographical restraints than pumped hydro or

compressed air energy storage [2, 10-13]. LAES is described in the following section.

2.3. Liquid Air Energy Storage

Cryogenic energy storage (CES) or liquid air energy storage (LAES) is a method of storing energy by liquefying a gas, usually air [13]. They use mechanical work to drive an air liquefaction process, the energy can be utilised at off peak times from renewable energy sources to help with their intermittency problems. The liquefied air is then stored in an insulated vessel until the energy is required again, at which point the liquid air is passed through a heat exchanger where it's evaporated. The resultant high-pressure gas is then used to drive an expansion device. The process is shown in Figure 2.4.

The key advantage of LAES over pumped hydro and compressed air energy storage is the much smaller geographical area it requires [14]. The liquid air is simply stored in insulated cryogenic tanks or cold storage substrates, which can be setup in almost any geographical location. It is reported that LAES currently has a very high investment cost. Compressed air energy storage has been developed for significantly longer time (since 1969 [15]), and the technology is considered mature with low cost. Furthermore, compressed air energy storage and LAES share a lot of the same components. Therefore, it could be assumed that utilising the mature CAES components for LAES could significantly reduce the investment cost. The biggest advantage LAES has

over battery energy storage is the much-reduced cost and increased longevity, mainly because the storage medium is air, which is abundantly available.

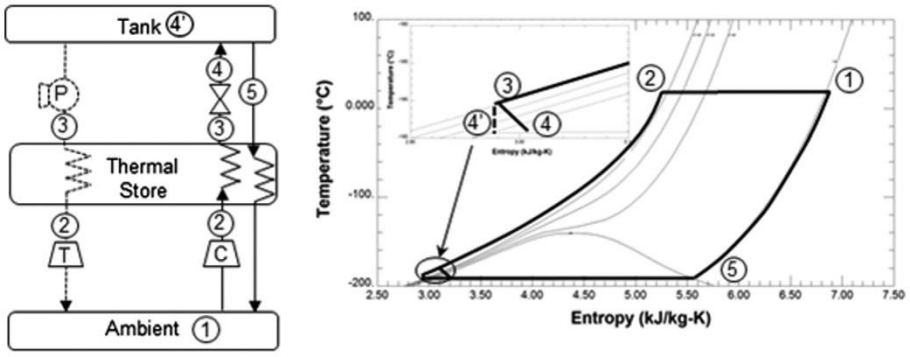


Figure 2.4 – Simple LAES cycle (a) process diagram and (b) TS diagram [13]

One of the first mentions of LAES in literature was in 1977, where Smith [16] reviewed its possibility, however the idea lacked significant development for many years. The next noteworthy research was in 1997 from Ordonez and Plummer [17], who studied the performances of different substances used for cryogenic energy storage (CES) system cold reservoirs. Their conclusions showed that even though some gases had better specific energy than air and nitrogen, these two would still be the best option as they have high availability. Chino and Araki [18] came up with the novel idea of using liquid air in a gas turbine's combustor, replacing the conventional air intake, which allowed a much greater inflow of air and therefore more fuel to be combusted. When compared to an ordinary gas turbine, they found the power output would be almost doubled. Li et al. [19] analysed a similar concept, where the liquid air was split into liquid oxygen and liquid nitrogen. The combustor would use the oxygen whilst the nitrogen's cold energy would be first utilised in CO2 separation, after which it would be super-heated by the gas turbine

exhaust and then passed through an expansion device to increase power output.

Using liquid air as the working fluid in an open Rankine cycle has also been explored [20], this method produced an efficiency comparable to 'hydrogen storage' and compressed air energy storage systems. The Rankine and Brayton cycles were both considered as candidates to utilise a cryogen's cold energy by Li et al. [21]. The Rankine cycle was found to perform better with low grade heat sources and the Brayton cycle did better when coupled with high grade heat sources. For the best performance, the choice of cycle should be made depending on the available heat source.

Figure 2.5 – Highview pilot LAES plant, UK 2010 [13]

The practicalities of LAES were assessed through the design and implementation of a pilot grid-scale LAES plant by Morgan et al. [11-13], Figure 2.5 shows the plant installed by Highview in Slough. The measured round-trip efficiency of the plant was 8%, the small size of the plant was given as the reason for the low value. The LAES was later further analysed by Guizzi et al. [3], who theorised that integrating the storage and recovery

systems allowed a round-trip efficiency of 54-55% to be achieved. Evans et al. [22] published that any CES system (including LAES) could only achieve a maximum round-trip efficiency of 40-50% due to the liquefaction process requiring large amounts of energy. The inefficiencies of liquefaction was acknowledged by Ding et al. [23], however they point out the longer lifespan of CES systems compared to hydrogen and battery systems plus the significantly reduced geographical footprint when compared to compressed air and pumped hydro energy storage systems.

A thermodynamic and economic model by Legrand et al. [24] using a LAES system with packed bed cold storage which was predicted to produce a round trip efficiency of 51.7% when integrated in the Spanish power grid. When integrated with the output of photovoltaic solar power plants to level the power between day and night, the LAES system can be cost competitive with compressed air and pumped hydro energy storage systems. Vecchi et al. [25] investigated the performance of a LAES system when operating at off-design conditions, as commonly seen in real power plants, finding that the round-trip efficiency could vary up to 30% from the predicted optimal.

So far in literature, real LAES systems have shown fairly low efficiency when operating as stand-alone systems, for this reason many have looked at the possibility of combining LAES systems with other applications that make use of the wasted cold thermal energy. Damak et al. [14] stated that LAES has much better efficiency when combined with other systems to make use of the cold energy. Using LAES as a backup for compressed air energy storage

systems was proposed by Kantharaj et al. [26]. They found that it would have a slightly increased round-trip efficiency compared to LAES alone. A further concept from Li et al. [27] was the integration of solar technology with an LAES system to provide increased power output through super heating the liquid air and passing it through an expansion device. When compared to the total power output of both the systems running separately, an increase of 30% was found from their theoretical study.

Antonelli et al. [10] modelled various hybrid LAES configurations and found that an equivalent round-trip efficiency of up to 80% could be achieved. Khalil et al. [4] created system models for two different systems integrating the expansion and liquefaction cycles, finding a maximum round trip efficiency of 84.15%. An integrated LAES and thermochemical energy storage system was proposed by Wu et al. [9], they found this configuration had a round-trip efficiency that was 13.3% higher than stand-alone thermochemical energy storage and had an energy storage density 3.4 times larger than LAES alone. Zhang et al. [28] tackled the problem of energy wasted through compression heat by incorporating a cascade style compression system and an energy recovery system in the form of an organic Rankine cycle or Kalina cycle. With these modifications they found the round-trip efficiency could be improved by 10.9-19.5%.

The cost and performance of LAES and pumped thermal energy storage was compared by Georgiou et al. [29, 30], where it was concluded that both systems were comparable in terms of cost and performance. However, they

did mention that the compressors and expanders are the most expensive components and that achieving the estimated performances of both systems are highly dependent on the development of low cost, high performance compressors and expanders.

2.3.1. Small-Scale LAES

Small-scale CES/LAES has received relatively little development, although some possible applications have been explored. Ordonez [31] investigated automotive applications of the Brayton cycle utilising cryogen cold energy, finding that its main limitation was the driving range, which was proportional to the cryogen's specific energy. Ahmad et al. [32-34] presented a combined residential air conditioning and power generation system utilising liquid nitrogen, finding that it gave a saving of 36% over conventional air conditioning systems, and could be increased to a saving of 81% if the cost of liquid nitrogen was reduced. Many of these investigations claim an improved cycle efficiency if the cost of cryogens were to be reduced and therefore the gas liquefaction process should be the focus of development.

Another application that would benefit from a reduction in the cost of liquefied gas is Dearman's liquid air engine. The Dearman engine boils liquefied gas (nitrogen or air) using a liquid heat exchange fluid, which is mixed with the cryogen inside the engine. The expansion from the boiling cryogen is used to produce work. The method using heat exchange fluid removes the need for large heat exchangers making the technology much

more compact [35]. Dearman have integrated the engine design into a truck refrigeration unit, where the Dearman engine produces the work to drive a vapour compression refrigeration cycle and the waste cold energy is directly used for increased refrigeration, giving improved efficiency.

The Dearman engine has the following advantages when compared to battery technology [35]: a lower capital cost, lower cost for increased storage capacity (as only the tank size needs to be increased), a very short recharge (or refuel) time, a competitive energy density, and a long working lifetime with easy maintenance and no degradation of performance over its lifetime. Whilst this technology would benefit from low-cost gas liquefaction, it would also benefit from the development of small-scale gas expansion devices that is described in later sections.

2.4. Liquefaction of Air/Nitrogen

The largest proportion of exergy destruction (efficiency loss) in LAES is found within the charging system (liquefaction system) and specifically within the compressors and expanders [36]. Due to this, the focus of this project was turned to this area.

The majority of gas liquefaction is currently carried out at industrial scale plants for nitrogen, oxygen and liquefied natural gas (LNG). Oxygen and nitrogen are liquefied mainly for medical, research and refrigeration applications. These industrial scale plants frequently utilise the Claude liquefaction cycle or variants [37], where liquefaction is achieved through

repeated compression, cooling and expansion of the gas. To assess the performance of a liquefaction cycle and allow for fair comparison between different systems, the Figure of Merit (FOM) is used. FOM for a liquefaction system is defined by Barron [37] in Equation 2.1, where w_{ideal} is the ideal work per kilogram required to liquefy a gas and w_{actual} is the actual work required.

$$FOM = \frac{w_{ideal}}{w_{actual}} \tag{2.1}$$

The ideal liquefaction system is a theoretical system involving two processes: isothermal compression of a gas to a very high pressure and isentropic expansion back to ambient pressure, where it will be in the saturated liquid state. The ideal work requirement can be calculated using Equation 2.2 [37], where T_1 is ambient temperature, s_1 and s_f are specific entropy values for ambient and saturated liquid conditions respectively and h_1 and h_f are the specific enthalpies for ambient and saturated liquid conditions respectively.

$$w_{ideal} = T_1(s_1 - s_f) - (h_1 - h_f) \tag{2.2}$$

2.4.1. Cycle Analysis and Modelling

A few publications analyse air/nitrogen liquefaction cycles, these provide some insight into the current state-of-the-art. The following literature focussed on the Linde-Hampson, Claude, Kapitza and Collins liquefactions cycles. The Linde-Hampson cycle is the simplest, involving only a compressor, a heat exchanger, an expansion valve and a liquid/gas separator. The Claude

cycle builds on the Linde-Hampson cycle by adding a gas expander and two additional heat exchangers. The Collins and Kapitza cycle are variants of the Claude cycle, with the Kapitza cycle being simpler and the Collins cycle being more complex. These cycles are described in more detail in Chapter 3.

An analysis of the Kapitza liquefaction cycle was presented in 2014 to find its optimal operating conditions [38]. Using ideal isothermal compression and an expander efficiency of 80%, the maximum FOM found was 0.78. Abdo et al. [39] later compared the Linde-Hampson, Claude and Collins liquefaction cycles for use in CES applications. The Claude and Collins cycles were found to perform best, with similar results. The lower cost of the Claude cycle led it to be recommended.

Analysis of the Claude and Kapitza cycles using commercial software was presented by Borri et al. [40], who concluded that the use of two-stage intercooled compression could increase cycle performance by 25%. They also found the third heat exchanger in the Claude cycle to be inefficient and recommended its removal. Lastly, operating the phase separator at higher than ambient pressures led to a performance increase. However, this means the liquefied gas would have to be stored at this same pressure, meaning a sealed pressure tank would be required. The Kapitza cycle with two-stage compression had the best overall performance, with operating conditions of 38-45bar compressor outlet pressure and 6bar phase separator pressure. This resulted in a specific energy consumption of 1872-2016 kJ/kg. Using Equations 2.1 and 2.2 the corresponding FOM values would be 0.32-0.35.

2.4.2. Current Small-Scale Gas Liquefaction

The first mention of a small-scale nitrogen liquefier is from Collins [41] in 1955, who described, designed and tested liquefier based on the Claude cycle. From the values given, this liquefier showed a figure of merit (FOM) of approximately 0.16, using a compressor pressure of 13bar. As of 2020, a number of small-scale liquid nitrogen generators are on the market [42-46]. These generators are primarily purchased by research institutes and hospitals. These generators do not have very high efficiency, with approximate FOM values ranging from 0.014[42] to 0.14[45], for generators producing 0.2 to 20 litres per hour respectively. Most of these generators are said to use the Gifford-McMahon cryocooler process to generate the liquefaction temperatures. The Gifford-McMahon cryocooler utilises a periodically switching valve (to alternate the pressure from high to low state) with a displacer/regenerator to transfer the cooling gas to the correct section. The cycle can be seen in Figure 2.6. Radebaugh [47] stated that Gifford-McMahon cryocoolers have become popular due to their low cost and high reliability. However, for small-scale energy storage applications the liquefaction FOM should be higher.

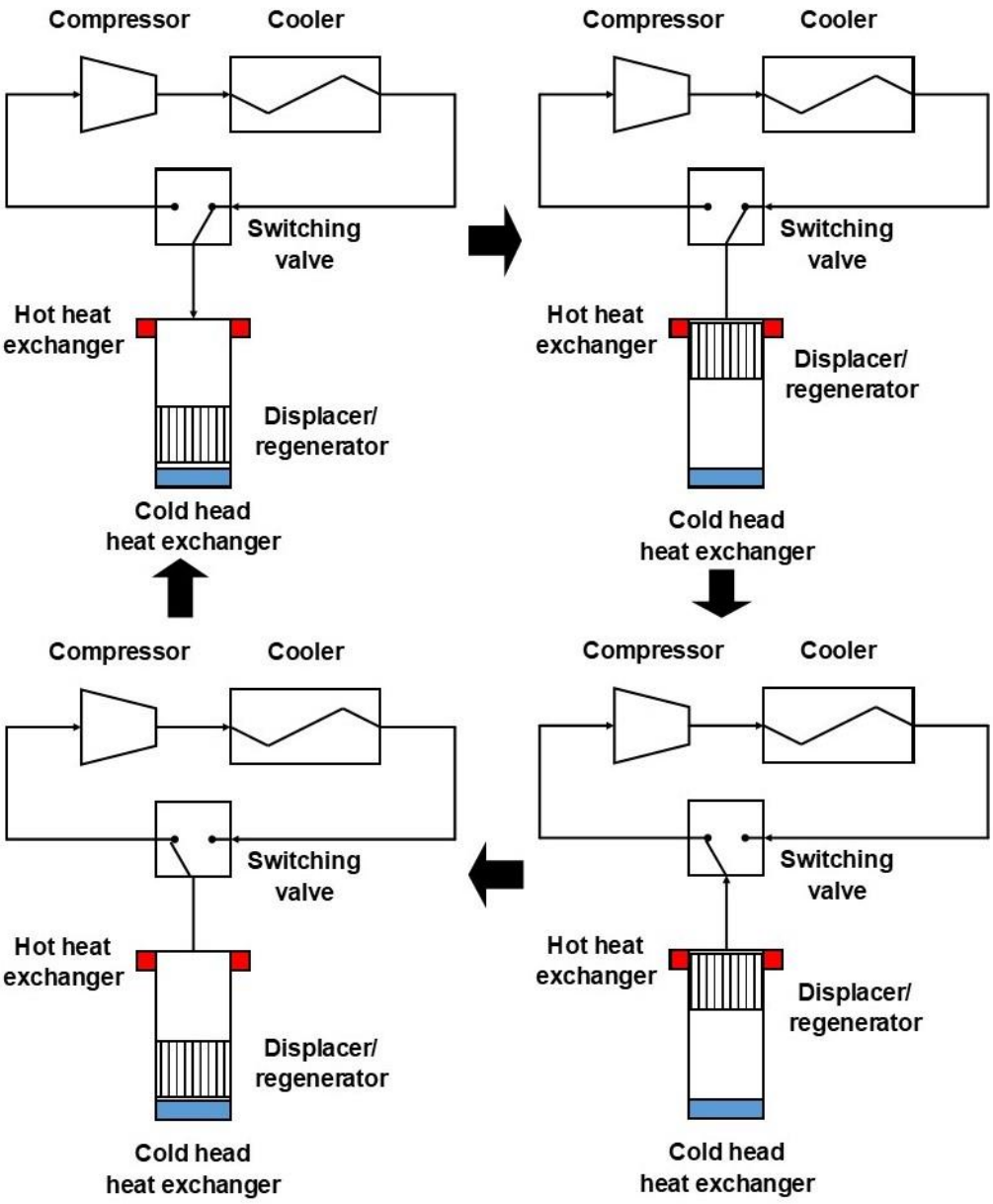


Figure 2.6 – Gifford-McMahon cryocooler cycle diagram

The reason natural gas is liquefied is to reduce the size of containers when its transported and therefore the cost, plants generally use cascade or mixed refrigerant cycles [48]. These cycles are more efficient than those without refrigerant and have the advantage that the fluid to be liquefied does not need to be compressed and therefore, no lubricant oil needs to be separated [49, 50]. These benefits could make the cycles appropriate for small-scale air

liquefaction applications. Cascade cycles have multiple different cooling circuits each with different refrigerants for different cooling temperatures, this has the disadvantage of needing multiple compressors and other components, which can lead to a large and expensive system. Mixed refrigerant systems are similar to the cascade cycle, but with the multiple types of refrigerant mixed together in one cycle, instead of in separate cycles. This means components are only required for one cycle, making it smaller, simpler and cheaper. The main disadvantage is that the system can only reach temperatures as low as the mixture's freezing point, but this does not tend to be a problem for air or nitrogen. Other disadvantages are the lack of thermodynamic property data for refrigerant mixtures and the tendency for heavier refrigerants to accumulate in certain sections of the system. Wang et al. [51] designed and analysed a small-scale mixed refrigerant nitrogen liquefier, which achieved a FOM of 0.245 with nitrogen entering at 8bar. Finally, Nguyen et al. [52] stated that mixed refrigerant cycles are more efficient than expander based cycles, over a larger range of conditions, but also more complex. From these reports, the mixed refrigerant or cascade cycles could perform well, but at a much greater complexity, initial cost and maintenance cost which could discourage investors.

The many possibilities of liquid air in energy storage application were reported by Strahan et al. [2], one of the conclusions states that efficient, cost-effective small-scale liquefiers are necessary for further development. The advantage of a small-scale liquefier is it can be located on site, eliminating

the need for liquid air delivery. This results in less expense in delivery and logistics plus reduced pollution from transport. Furthermore, it could increase the efficiency of on-site renewable energy sources (wind and solar) by utilising any unused energy. It is also pointed out that LAES could be used to replace the countless diesel generators which provide backup power for buildings.

2.5. Organic Rankine Cycle

Another emerging technology that shares a lot of the same components as CES/LAES systems is the organic Rankine cycle (ORC). The ORC is a system that utilises the principle of the Rankine cycle, but instead using organic fluids such as refrigerants. As it used the same principle of the Rankine cycle, the components it consists of are much the same, comprising of a pump, an expander, an evaporator and a condenser as seen in Figure 2.7. The main advantage of the ORC is it can utilise heat sources at temperatures of 60-350°C, which is much lower than steam Rankine cycles, which generally require heat sources at 500-600°C [5]. This allows the ORC to be utilised in many applications where low-grade heat is available or may otherwise go to waste. Some of these applications include:

- Biomass-fuelled combined heat and power [5, 6]
- Geothermal [5]
- Concentrated solar systems [6]
- Solar salt ponds [6]
- Solar reverse osmosis systems for desalination [6]

- Internal combustion engine waste heat [5, 6]
- Ocean thermal heat sources [5, 6]
- Industrial processes waste heat [6]
- Gas turbine waste heat [5]
- Steam cycle waste heat [5]
- Cold production by driving a refrigeration cycle compressor [6]

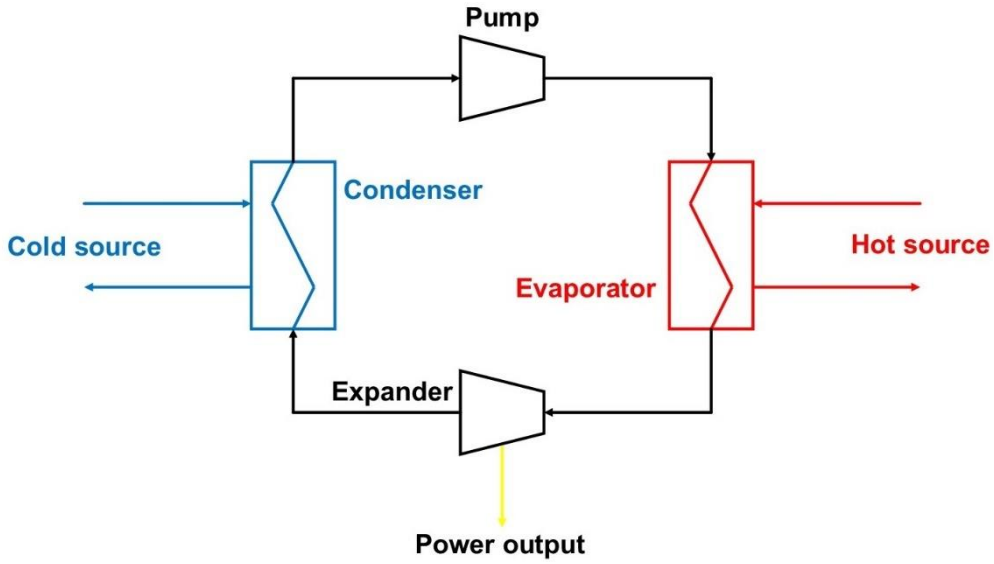


Figure 2.7 – Simple ORC diagram

Quoilin et al. [6] stated that the working fluid and the expansion device are the two key aspects in an ORC system but noted that experimental studies on certain types of expander were scarce. Furthermore, they stated that calculation methods for expander efficiency in literature often varied, making it hard to compare between publications and with some values being misleading. Rahbar et al. [5] also stated that there were few studies on the design, modelling, and experimental testing of expansion devices, even though they are critical to the ORC efficiency. They follow this up by stating

that high expansion ratio and efficiency expanders should be developed for ORC development.

2.5.1. Small-Scale ORC

If the ORC is used in small-scale distributed locations, it can provide further benefits. Approximately, 12% of grid electrical power is lost in the transportation of it [5]. Therefore, generating power at the place of use would remove this loss. It would also provide other advantages such as using it as backup power, less disruption in natural disasters, environmentally friendlier, better for remote areas, more security, and more reliability. To help make small-scale Rankine cycles attractive to investors, it needs to be competitive in terms of performance and cost. As expansion devices are clearly key to the efficiency and often the most expensive component, there is a need to develop a low-cost high efficiency expansion device.

2.6. Small-Scale Expansion Devices

Before different types of expansion devices are reviewed, several authors in literature have given advice on the best practices and problems with comparing expansion devices. The first (and most obvious) parameter to compare expansion devices is their performance. 'Performance' generally refers to the power output and efficiency of an expansion device. Multiple sources mention that the dimensionless parameters of specific speed and diameter of expanders can be used to select an expander type [53-55]. These dimensionless parameters can be used to plot efficiency lines for different

expander types on a Balje diagram, as shown in Figure 2.8. However, as stated by Zywica et al. [56], performance shouldn't be the only basis for selection. If an expander needs frequent repairs or it is excessively noisy it would not be received well by the user.

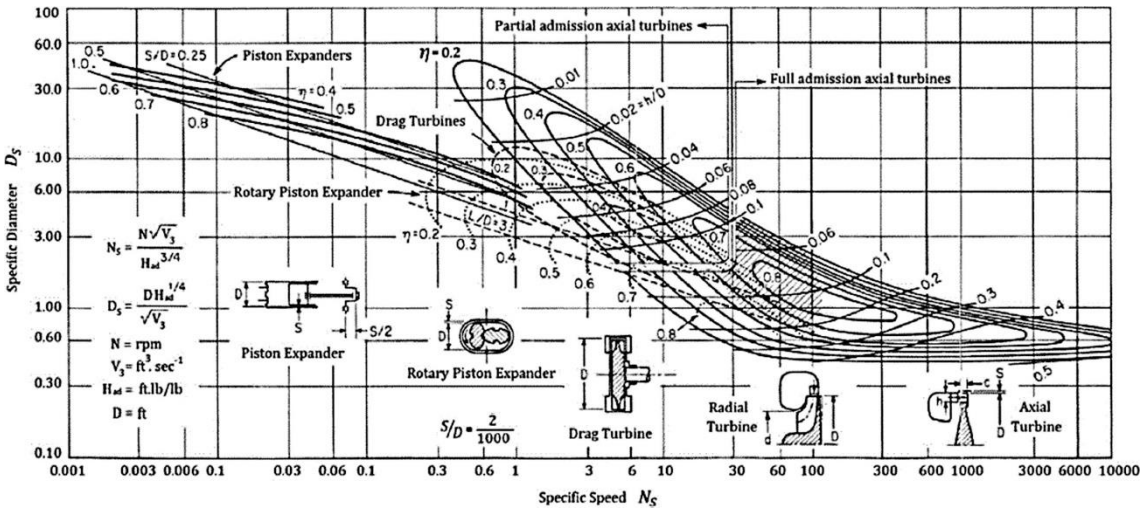


Figure 2.8 - Balje diagram for selecting expander/compressor devices [55]

According to Imran et al. [54] the choice of expander depends strongly on the operating conditions, working fluid and size requirements of a system. Dumont et al. [57] explains that the main criterion for expander selection is the system power range, stating that screw is best for above 10 kW and scroll or piston are best for less than 2.5 kW, although this study excluded turbine expanders. They go on to point out that there is a problem with directly comparing different expander types due to their different technological maturities, different designed operating conditions/fluids and test rig limits (if using the same test rig). Vittorini et al. [58], concerning compressors, says the technology on the market is not aligned with best performance standards as old designs are used with no evidence of upgrading. As many expander

designs are based on pre-existing compressor designs, this problem could be carried over.

The second most important criterion is the expansion ratio required by the system, followed by the required lubrication and expander cost [57]. Lemort and Legros [59] pointed out that selecting an expansion ratio lower than ideal is preferable as it does not affect the performance much (a 20% reduction in expansion ratio results in a 2% reduction in efficiency). Finally, device complexity should be accounted for, for example, devices without valves are simpler and have increased reliability.

2.6.1. Turbine Expanders

Turbine expanders utilise the velocity of the fluid to create power output. A set of stator blades or nozzles are positioned before each rotor wheel which serve to increase and direct the fluid velocity. The rotor wheel then either converts the kinetic energy to velocity of the rotor (impulse turbine) or utilises the kinetic energy and pressure differential to move the rotor (reaction turbine), Figure 2.9 demonstrates this.

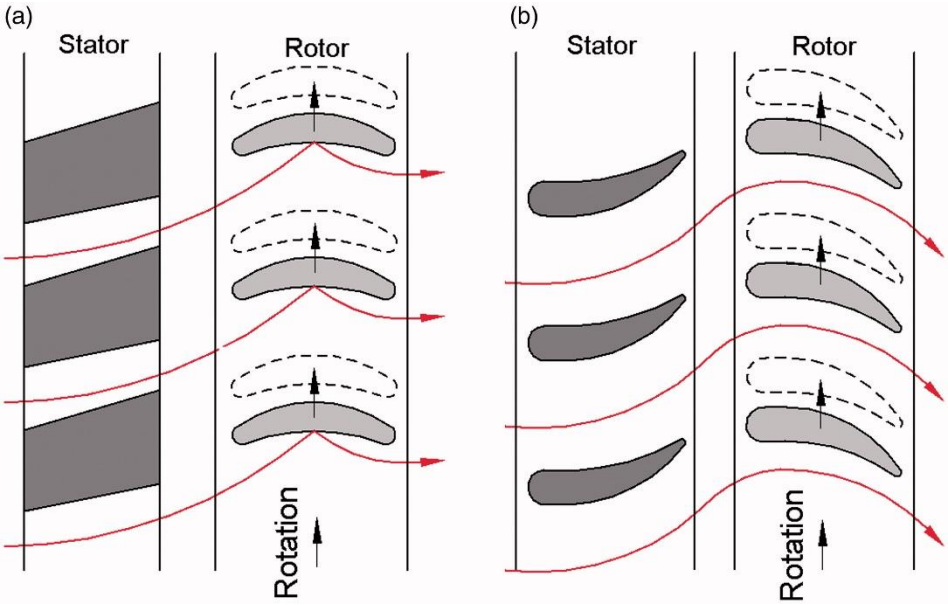


Figure 2.9 – Comparison of (a) an impulse turbine and (b) a reaction turbine [56]

Advantages

It is popular opinion that turbines are best for systems above 10kW [53]. Their high flow rates and operating speeds result in a large power to size ratio, making them very compact devices compared to other expanders. They have low vibration and noise, as they are intrinsically balanced plus their operation is simple as they have a single moving part. This results in little wear and high durability and reliability [56]. Turbines do not require any lubrication mixed with the working fluid, as there is no contact between moving parts [60]. It is possible to run very high expansion ratios through them if needed, however, at very low efficiency. This is because a larger proportion of the pressure will not be used to drive the turbine. They can also be altered very easily for different operating conditions by changing the nozzle ratios, blade heights or degree of admission [60].

Disadvantages

Designing an efficient and low-cost turbines with power outputs less than 1kW is 'very difficult' according to Zywica et al. [56]. In fact, Badr et al. [53]and Qiu et al. [61] both agree that at power outputs less than 50kW, turbines have low efficiency and high cost. Furthermore, the speed and cost of turbines create problems for systems under 50kW [54, 61]. This is mainly due to the high relative surface roughness of the blades, large relative clearances between blades and casing and a relatively much smaller trailing edge thickness of the blades [60]. Saghlatoun et al. [62] agree saying that all losses are much higher in small-scale turbines. Weiß [60] states that leakage is inevitable in turbines and Zywica et al. [56] goes on to say that the leakage above the blades is the main reason for power loss in turbines.

Turbine expanders' inability to handle two-phase flows prove to be one of their largest disadvantages, with multiple sources stating that it leads to erosion of the turbine blades and a much reduced operating life [53, 56, 60, 62]. This means that the control of the attached system should be much greater, to ensure two-phase conditions do not occur, increasing the cost of the system. Some claim that turbines with rotational speeds up to 60,000rpm present manufacturing cost problems due to the high forces and also requires a reduction gearbox to connect to an electrical generator [62]. Others state that turbine speeds can reach 100,000rpm [60], which would clearly exacerbate this problem. Finally, turbines are not good with unstable conditions [62],

such as may be produced by a pulsing compressor or dynamic valves in an attached system.

Performance

Although turbines in the megawatt power range can achieve efficiencies of above 90%, most micro turbines have efficiencies below 80%. This is backed up by the fact that even though a CFD simulation predicted an efficiency of 88% for a 30kW turbine, the actual measured efficiency was 75% [56].

Axial or radial flow turbines

Axial flow turbines are better for applications where a higher flow rate and lower pressure drop are required [56]. If impulse axial turbines are used, expansion ratios greater than 100 are possible in supersonic conditions, whilst maintaining efficiencies of 70-80%. They are also easier to manufacture [56, 60] and easier to implement a multistage device than radial [56].

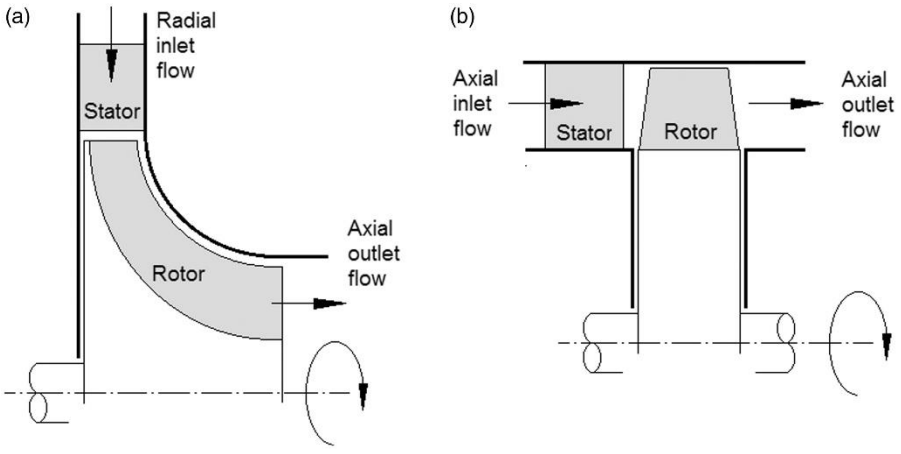


Figure 2.10 – Cross-sectional views of (a) a radial turbine and (b) an axial turbine [56]

Radial flow turbines are generally considered more suitable for small-scale applications [60, 62] and handle lower flow rates much better [56, 60]. They also display better efficiency in low/medium temperature systems, such as ORCs and refrigeration [56]. Radial flow turbines are cheaper than axial flow turbines and have a higher stiffness, so have a better dynamic performance [56, 62]. They can also handle higher pressure drops, when flow is not supersonic and are less sensitive to changes in operating conditions [56]. A comparison of radial and axial turbines is shown in Figure 2.10.

2.6.2. Volumetric Expanders

Volumetric expanders utilise various methods to trap fluid in a chamber that changes volume. There are many designs, the most popular of which are described in the next sections, along with their individual advantages, disadvantages, and performances. However, the following advantages and disadvantages apply to all volumetric expanders when compared to dynamic (turbine) expanders.

Advantages

Volumetric expanders perform best in applications below 10kW [53] and are frequently found in applications below 100kW [56]. They are generally considered much better for low flow and speed applications [54, 57, 60, 61, 63] and for higher pressure drops [54, 57, 63] (excluding the supersonic turbine case mentioned in the previous section). Volumetric expanders are much cheaper than turbines [54, 63].

They can generally be connected directly to an electrical generator, with no need for a reduction gearbox [60]. Volumetric expanders on the whole can accept two-phase flows as the velocity of the fluid is not high enough to cause erosion [54, 57, 61]. Finally, compressors from refrigeration systems, which can be reversed and used as expanders [60], are readily available.

Disadvantages

All volumetric expanders have a single optimum expansion ratio, if a system does not operate at this expansion ratio then under/overexpansion will occur, resulting in a loss of efficiency [54, 57, 59]. However, heat transfer and leakage losses can also result in overexpansion [59], which has the larger effect on efficiency.

Volumetric efficiency (or filling factor) decreases with speed due to pressure drops over inlet/outlet ports and valves, often limiting the speed. However, the volumetric efficiency value alone cannot determine the extent of the pressure drop effects, as leakage losses will increase the value, possibly above unity at low speeds [57]. In this case, more gas would be leaking from the inlet to the outlet (per cycle), than the swept volume of the expander's 'chamber'.

Many volumetric expanders have contact sealing mechanisms, which leads to increased friction, wear and leakage loss problems [54, 56, 60]. To reduce this issue, lubrication is often required, the inclusion of which creates a more complex device and can potentially cause problems with the attached system.

Zywica et al. [56] also mentions that the bearings of volumetric expanders tend to have reduced lifespans due to the oscillating loads and vibrations created in operation.

The market for volumetric expanders is far from mature, only a few are currently available. This makes them much more expensive [54, 61]. Volumetric expander designs generally have more complex designs, which make them harder to assemble, more likely to fail and have higher mechanical losses [54]. Finally, Weiss [60] states that the single stage is generally limited to and expansion ratio below 10 and, although multiple stages are possible, it is much more complex than with turbines.

2.6.3. Reciprocating Piston Expanders

Reciprocating piston expanders are based on the technology in the widely used piston combustion engines, wherein a piston moves back and forth within a cylinder. A piston ring forms a contact seal between the piston and cylinder and with one end of the cylinder being sealed off, the volume is thus changed. The working fluid is controlled with inlet and outlet valves, as shown in Figure 2.11. The contact seal is usually lubricated with oil via splash or injection methods. The piston is connected to a crankshaft with a connecting rod, allowing the reciprocating motion to be converted to rotational motion for the output shaft.

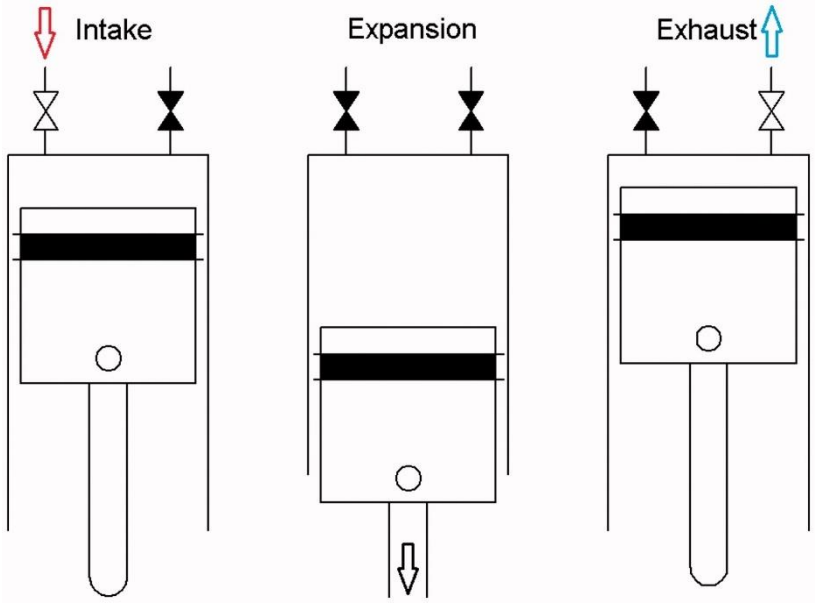


Figure 2.11 – Working principle of a reciprocating piston expander [56]

Advantages

Reciprocating piston expanders are good for low flow and low power applications [57]. They can have the highest expansion ratio of the volumetric expanders [54], as the ratio can be controlled by increasing the stroke length of the piston. Lemort and Legros [59] goes on to state that they can also handle the highest pressures and temperatures of the volumetric expanders. They can achieve relatively high efficiencies [54], as they can be sealed well and controlled precisely with inlet and outlet valves, however, this requires that the mechanical losses of seals and valves to be not excessive.

Saghlatoun et al. [62] state three further advantages. Firstly, they run at a low speed, so do not require reduction gearboxes for a generator. Secondly, they handle unstable conditions well and can adjust to a range of operating conditions. Lastly, it is stated that piston expanders can handle two-phase

flow. However, Weiß [60] disagrees and states that liquid in the cylinder can cause damage.

Disadvantages

Zywica et al. [56] disagree with the statement in the previous section and say that piston expanders generally have low efficiency due to their large mechanical losses. The fact that they have the most complex structure assembly [54, 56, 60, 62] and require a valve setup [60, 62]. This factor also leads to them being heavier than other expansion devices [62] and more expensive [56, 60, 62].

Piston expanders create torque pulses that will increase mechanical fatigue on mechanical devices that are attached [62]. Related to this they also create flow oscillations [54, 62] which can detrimentally affect the connected system/components. Due to their reciprocating motion, piston expanders require balancing [54, 60, 62]. However, even with this balancing they still experience the highest noise and vibrations [56, 60, 62]. Finally, despite good lubrication methods being developed, they still suffer high frictional losses [56, 62]. This along with the large part count means they have low durability [62].

Performance

Imran et al. [54] state that the maximum isentropic efficiency in literature is 76%, but most reside below 50%. They successfully utilised a semi-empirical model with a maximum error of 4.7% compared to experimental results.

Saghlatoun et al. [62] and Weiß [60] state a similar figure of 70% for a maximum efficiency of the piston expander. There appears to be some discrepancy as to the achievable expansion ratio, with Weiß [60] claiming they can't exceed 10 but Lemort et al. [63] stating that up to 14 is possible.

2.6.4. Screw Expanders

Screw expanders can be split into single and the more popular twin-screw variety. The single screw expander has one screw shaped rotor and two gate rotors which mesh into the screw's grooves. The gate rotors seal off parts of the screw grooves and as the screw rotates these sealed sections change in volume, this is demonstrated in Figure 2.13 [64]. Twin screw expanders operate on a similar principle, except the job of the gate rotors is achieved by a second parallel intermeshing screw, a double screw expander is shown in Figure 2.12. The gas enters the screw threads on one side when the volume between the threads is minimum. Then as the screws rotate, the volume increases until a maximum value, at the opposite side of the expander.

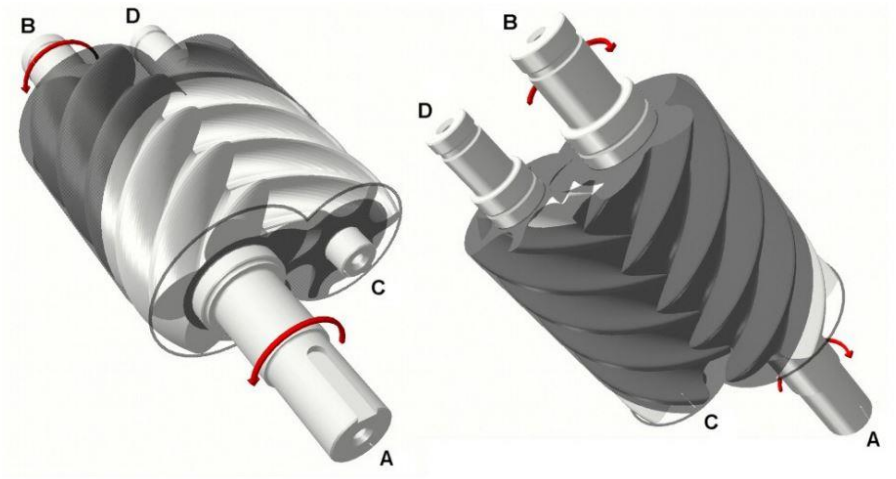


Figure 2.12 – Double screw expander view from above and below [61]

Advantages

Lemort and Legros [59] claim that screw expanders have the highest power outputs and efficiencies of the volumetric expanders and Dumont et al. [57] back this up by pointing out their high speed operation makes them compact with a high power to size ratios. They do not require valves which provides multiple benefits [60, 62] such as reduced complexity and mechanical losses. Screw expanders are said to have the greatest technical maturity [62, 63] as they have been used extensively for many years as compressors and can be easily converted. This also helps them to be 40-60% cheaper than turbines [62].

One of the most cited advantages of the screw expander is its ability to handle two-phase flow [56, 57, 59, 60, 62, 63]. Wet expansion can improve the performance of screw expanders as droplets can sometimes provide lubrication and help seal leakage gaps [54, 56]. However, wet expansion can also cause the deterioration of lubrication, leading to increased friction and wear [56].

Disadvantages

Screw expanders are not readily available for applications under 10kW [61]. This is probably due to leakage being larger in small-scale devices, therefore, they are not viable for applications less than 25kW [54, 61]. In order to reduce leakage screw expanders generally rely on small clearances between moving parts, some claim clearance dimensions of 20-50μm are used [54], whilst

others claim that 50µm is the most common clearance size/shape accuracy [56, 61, 62]. Partly due to this fine clearance, which requires high tolerance machining, and partly due to the general complex geometry, manufacturing requires expensive equipment, leading to a large initial investment [53].

Another disadvantage in many screw expanders is the need for lubrication within the working fluid [60, 62]. The oil benefits the expander by reducing both frictional and leakage losses, however it is often detrimental to the attached system [59]. Only unsynchronised screw expanders require oil lubrication as the rotors contact each other, synchronised expanders do not, but they have added frictional losses from synchronising gears and increased leakage between clearances [54, 56], meaning this is only a solution if the system must not have oil present in the working fluid. Synchronised screw expanders can run faster as there are less hydrodynamic losses, but at the same time they need to run faster in order to reduce the increased leakages [59, 63]. The bearings of Synchronised screw expanders require more sophisticated lubrication and a more rigid casing, leading to further expense and complexity [56].

As previously mentioned, screw expanders are capable of high-speed operation, however, this presents some problems when attaching the output to a generator, with many requiring speed reduction gearboxes [53, 54, 60]. Due to the geometry of the screw expanders, the forces apply the large loads in both the radial and axial direction on the bearings [62]. Weiß [60] pointed out that at least four bearings are required (compared to a turbine's two),

however further bearings may also be required if the bearings cannot take both axial and radial loads.

Performance

There are many reports of different performance capabilities of screw expanders in literature. One of the highest claimed is 90% [56], however, this is probably for a large scale expander. Badr et al. [53] stated that screw expanders in the 5-50kW range have a maximum efficiency of 70%. Whilst Weiß [60] claimed a maximum of 50% with Saghlatoun et al. [62] backing this up with a tested case giving 53% efficiency. Imran et al. [54] gave a large range of 20-80% efficiency, for expanders in the power range of 1.5kW-1MW. They go on to say that isentropic efficiency increases with speed, but volumetric efficiency decreases, leading to an unfavourable trade off. Furthermore, they point out that leakage and friction losses are inversely proportional, when the clearance gap sizes are varied, offering another unfavourable trade off. Finally, the maximum expansion ratio for screw expanders is agreed to be 5 [59, 60, 63], meaning screw expanders will be used for lower pressure applications rather than piston expanders and multistage/supersonic turbines.

Single-screw and twin-screw expanders

Single-screw expanders are much less popular in literature than the twin-screw variant. Single-screw have more leakage than twin-screw and therefore have a lower theoretical maximum isentropic efficiency of 60% [54]. However,

they have the benefits of evenly distributed loads on the screw and a higher expansion ratios [56].

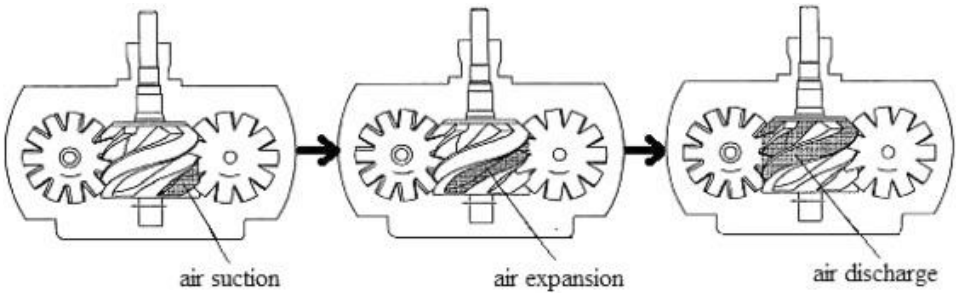


Figure 2.13 – Single screw expander working principle [64]

2.6.5. Scroll Expanders

Scroll expanders consist primarily of two plates each with a spiral shaped extrusion on it. When assembled, the scrolls mesh forming several separated volumes trapped within. One scroll is held stationary, whilst the other is permitted to translate in a circular motion. This motion moves the trapped volumes from the centre to the periphery and as this happens the volumes increase in size, as shown in Figure 2.14. Sealing is generally achieved through small clearances between the axial gaps and the flank gaps.

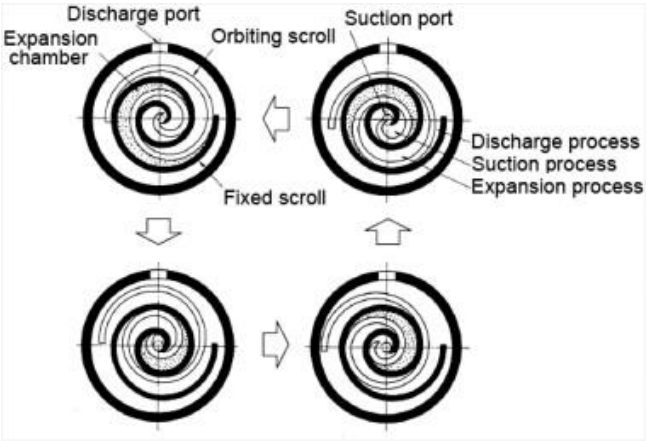


Figure 2.14 – Scroll expander working principle [61]

Advantages

Scroll expanders are said to perform best in the 0.1-10kW range of power outputs [59]. Zywica et al. [56] backed this up, claiming they can be used down to 0.1kW applications, however, Qiu et al. [61] stated that they are only commercially available in the 5-10kW range. In a study comparing volumetric expanders, the scroll expander had the highest efficiency for those below 5kW [57].

The scroll is a very simple assembly with a very low part count [56, 57, 62], the fact that it does not require valves helps this as well as reducing associated mechanical losses [62]. As well as the simple assembly, it has simple operation [56], both of these give the scroll expander high durability [62] and low cost [56, 62].

Like the screw, many acknowledge the scroll expander's ability to handle two-phase flows [59, 60, 62, 63], with Zywica et al. [56] also pointing out the two-phase flow can improve performance through droplets lubricating and sealing leakage gaps. The scroll expander is also said to have low noise during operation [62] and operates at a low enough speed to be mounted to a generator without a reduction gearbox [60].

Disadvantages

With the scroll expander, like other volumetric, leakage losses dominate at low speeds [54]. Lemort and Legros [59] points out that leakage between one chamber and another causes increased power in the scroll expander, as the

high pressure gasses leak into the low pressure chambers, however this reduces the efficiency of the expander. In order to minimise the leakages clearances between moving parts should be kept to a minimum, typically 20-180μm for the flank clearances and around 50μm for the axial clearances [65].

Some scroll devices have incorporated the use of contact seals for the axial clearances, however they require replacing at regular intervals [63]. Lubrication is often used in scroll expanders either to lubricate contact seals or help sealing clearances if none are used. However, although the lubricant greatly benefits the expander, it often has a detrimental effect on the attached system. Furthermore, [58] states around 5-7% of the power is required to pump the oil in lubricated expanders. Finally, Lemort et al. [63] claims that the temperatures in scroll expanders are often limited by the thermal expansion of the scrolls, due to their small clearances, and temperature effects on the lubricant.

Performance

The scroll expander is said to have the highest efficiency for expanders operating at less than 5kW, giving a maximum of 81% efficiency from an empirical model [57]. Imran et al. [54] and Lemort et al. [63] gave similar values of 83% and 87% isentropic efficiencies respectively. Qiu et al. [61] simply states that efficiencies above 70% are possible, with Weiß [60] and Ziviani et al. [65] both claiming that 70% is a maximum for small-scale. Zywica et al. [56] tells us that an isentropic efficiency of 80% was achieved in an experiment, but high noise and vibrations were observed. Whilst

Saghlatoun et al. [62] states that although a theoretical isentropic efficiency of 83% should be achievable, the maximum measured have been 30-50%. Finally, both Lemort et al. [63] and Weiß [60] agree that the maximum volume ratio of the scroll is 5, the same as for the screw expander.

2.6.6. Vane Expanders

Vane expanders consist of a rotor which can spin on its own centre axis, located inside an enclosing housing which is either circular or ellipse shaped. The rotor is generally offset so that there are varying distances between it and the outer housing at different points. Seals or vanes are used evenly spaced around the rotor to separate different volumes. The seals can move in and out of the rotor to account for the change in distance as it rotates. A cross-sectional view of a vane expander is shown in Figure 2.15.

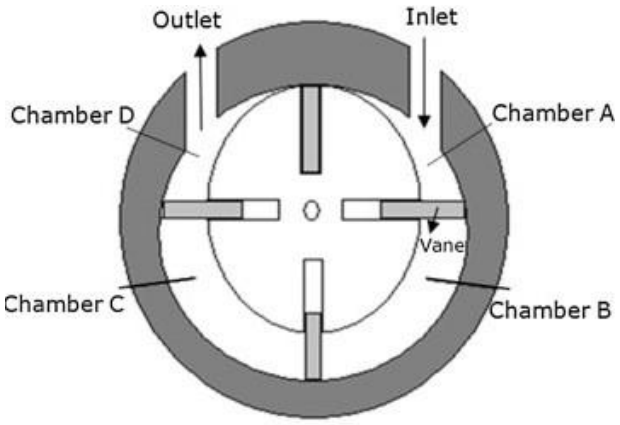


Figure 2.15 – Vane expander cross-sectional view [61]

Advantages

The largest advantage of vane expanders is their simplicity of design and manufacture [56, 62]. This simplicity means they have low maintenance [62]

and cost less than other expanders [56]. They are also relatively compact [56], can handle large expansion ratios and a large range of operating conditions [62]. As they operate at a very low speed, the do not require a reduction gearbox for connection to a generator [54, 62].

Disadvantages

The largest disadvantage of the vane expander are the friction and leakage problems [54, 56, 60, 62]. This is because the vanes require high force to seal effectively, however the forces on the vanes are not constant plus the large required forces create excessive friction. Vittorini et al. [58] stated that the friction is highly dependent on the speed, as the centrifugal force increases with the square of rotational speed but acknowledges that the leakage is also dependent on speed, increasing when lower. Aside from the large power and efficiency losses, the friction also creates problems with wear on the vanes and housing [60, 62]. Imran et al. [54] pointed out that revolving type vane expanders, where the outer housing rotates as well as the inner rotor, can significantly reduce friction problems as the relative velocities are much reduced.

It is well established that vane expanders require the injection of liberal quantities of oil for lubrication and sealing [56, 60, 61], with Vittorini et al. [58] claiming 5-7% of the power would be consumed by the oil pump. Although, Imran et al. [54] stated that development of low density vanes will allow much less lubricant to be used. Finally, vane expanders present

problems with adequate breathing [62] and are typically limited by a max temperature of 140°C [59].

Performance

The maximum efficiency found for vane expanders is 84% [62]. Weiß [60] found one report of 80% efficiency but said most lie in the region of 30-40%. Another source claimed the maximum to be 71% isentropic efficiency and 57% volumetric efficiency [54]. Both Qiu et al. [61] and Zywica et al. [56] claimed similar values at 54.5% and 50% respectively. Badr et al. [53] had the most pessimistic view, claiming vane expanders do not exceed 25-35%, however this is the oldest of the papers studied. Vane expanders are able to tolerate a maximum expansion ratio of around 8 [54, 59], making them second best in that area only to piston.

2.6.7. Rolling Piston Expanders

Rolling piston expanders consist of an inner piston rotor within an outer cylinder. The piston is permitted to roll around the inside of the cylinder in restrained motion. Usually one contact seal is located in a groove in the outer cylinder, that maintains contact with the rolling piston by means of a spring. This seal splits the volumes that are created by the gaps between the piston and the cylinder and allows the variation for expansion as shown in Figure 2.16.

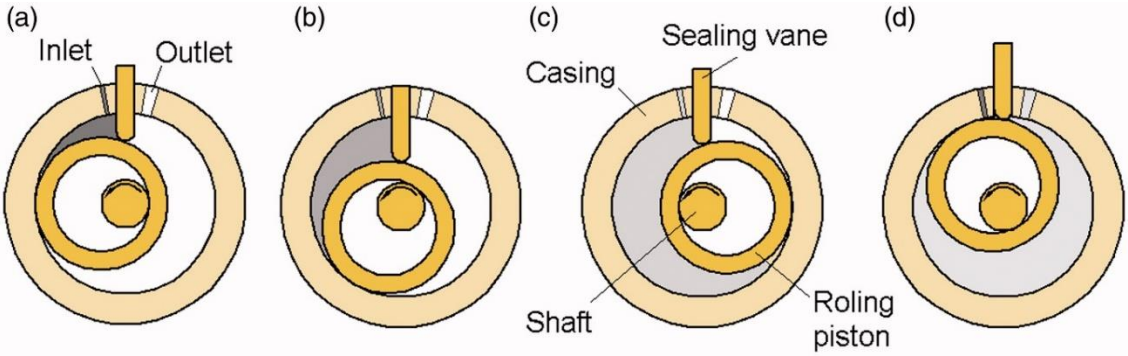


Figure 2.16 – Rolling piston expander working principle [56]

Rolling piston are not present much in literature, especially in comparison papers, However, they are said to be able to withstand high pressures, have a low cost, but also provide low expansion ratios, strong friction and problems with leakage [56]. Imran et al. [54] claimed they can produce isentropic efficiencies in the range of 40-60%, whilst Zywica et al. [56] said they do not exceed 40%.

2.6.8. Gerotor Expanders

Gerotor expanders consist primarily of two gear like components, an inner one which resembles an external gear and an outer one that resembles an internal gear with one extra tooth. Both components rotate about their own central axes, but with the axes offset from each other to allow the teeth to mesh. The gaps when the teeth are not meshed form a changing volume, whilst the point where they mesh is close to zero volume and prevents recirculation of the fluid. An example of a gerotor expander is shown in Figure 2.17.

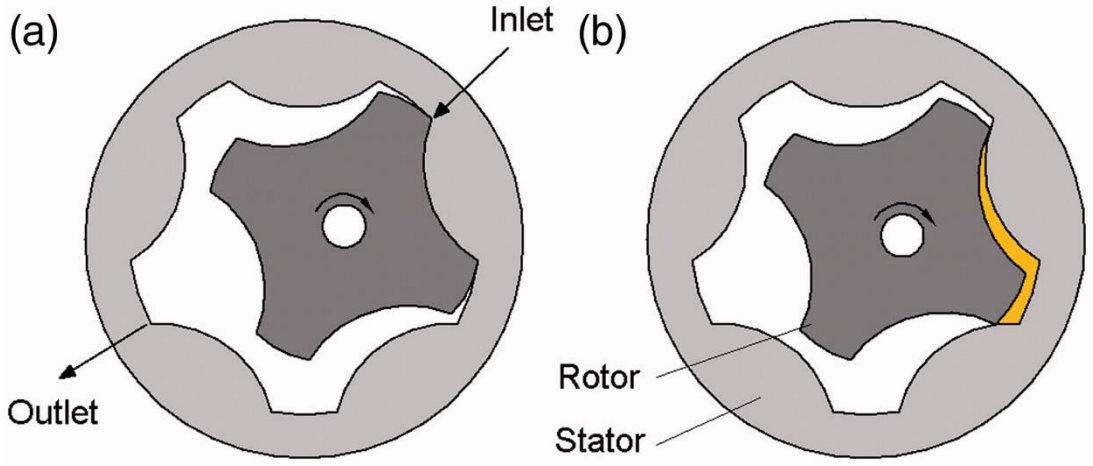


Figure 2.17 – Gerotor expander working principle [56]

Like the rolling piston, gerotor expanders have received little attention in literature and comparison papers. The design is simple, with only two moving parts [56, 62] and they subsequently have a low cost [62]. Gerotors also have less friction than other expanders, as the sealing contact areas have a rolling contact instead of sliding [56, 62]. Both Saghlatoun et al. [62] and Zywica et al. [56] state that the maximum efficiency found is 85% in lab conditions.

2.6.9. Expander Modelling/Simulations

Expanders are often modelled or simulated numerically to estimate their performance. It is therefore useful to know how accurate the models/simulations in literature typically are so that expectations of accuracy are correct. Imran et al. [54] gave values of 4.7% error in a semi-empirical piston expander model and 5% error in power output for a scroll expander model. For a screw expander model, they found a 10% error when using a polytropic model and 7.5% error when using an isentropic model. When they looked at a vane expander theoretical model a maximum 22% error was found and 25% when looking at a revolving vane model.

2.7. Wankel machines

2.7.1. Introduction to Wankel Devices

The first Wankel device was an internal combustion engine invented by Felix Wankel with the first prototype developed in 1951. The devices utilise a triangular shaped rotor which rotates on an eccentric rotating shaft, inside an epitrochoid shaped housing. When the rotor is placed within the housing, three volumes are created, separated from each other by the rotor's three apexes, which seal against the housing. The movement of the rotor within the housing causes the three volumes to continually change between a maximum and minimum value, this is what creates the compression, expansion and pumping required for the engine.

2.7.2. DKM and KKM

In one of his publications [66] Felix Wankel says that there are an almost infinite number of different rotary piston machine possibilities, following up by stating that the large number could lead to less development on any one specific type. However, saying this, since the inception of the Wankel engine, very little research and development has been done on other rotary piston engines. The standard Wankel engine that has primarily been developed is the KKM (Kreiskolbenmotor) type, however the first prototype produced by Felix Wankel was of the DKM (Drehkolbenmotor) type, a diagram of this is shown in Figure 2.18 [67]. Although the DKM type had a higher maximum speed and was better balanced, the KKM was chosen as it was more practical

as a combustion engine, especially for spark plug placement and general maintenance. The fact that the KKM was the focus of development in engines, could have been the reason the DKM type was not a popular development choice for compressor and expander technologies either.

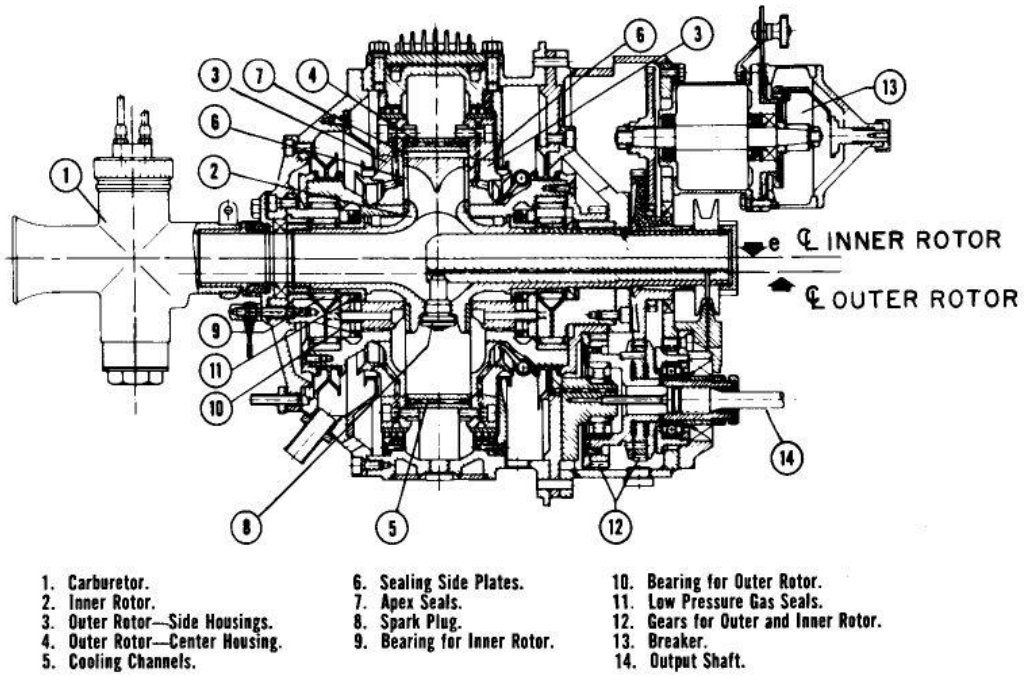


Figure 2.18 – Section view of DKM 125 Wankel combustion engine [67]

2.7.3. Wankel Components

The Wankel expander consists primarily of a triangular shaped rotor inside an epitrochoid shaped housing. In the standard type Wankel expander (similar to the KKM engine), the rotor rotates around its own central axis and simultaneously, that axis translates in a circular motion at three times the frequency of rotation. This motion allows the apexes of the triangular rotor to follow the shape of the epitrochoid housing as it rotates. The rotor motion transmits power to a crankshaft, on which it is mounted with a bearing. The

rotor motion is controlled with an internal gear mounted to the rotor and an external gear mounted onto the housing side. Typically, a Wankel engine has 3 apex seals, 6 side seals, 6 corner seals and a spring for each seal. Furthermore, the standard type requires balancing masses to be mounted to the crankshaft to eliminate the vibrations caused by the eccentric spinning of the rotor.

2.7.4. Producing the Shape

To produce the epitrochoid shape of the housing a pair of parametric equations are used to define x and y coordinates [68, 69]. The equations utilise two main parameters, the generating radius, R and eccentricity, e, the ratio of these two parameters is important because it defines both the volume ratio and the displacement of the expander. A trade-off exists when choosing this ratio where increasing R/e leads to a larger displacement but lower compression ratio. Ogura [70] claimed that an R/e ratio of 6 is best for a developed Wankel compressor, whilst [69] states that R/e ratios of 6-8 are best for combustion engines. Results from a study determined that varying the R/e ratio between 6-9 had negligible effects on the efficiency of a Wankel expander [55].

There is not much research on the width of Wankel expansion devices, Francesconi et al. [55] determined that width to diameter ratio ultimately did not affect maximum efficiency, just the operating conditions at which the maximum efficiency was achieved. Finally, Omirou and Nearchou [71]

researched a method for the machining of epitrochoid housings, however this is the only research paper found on this topic, theoretically leaving much room for improvement in the manufacturing, which could improve cost, surface finish and tolerance.

2.7.5. Calculations

Calculating the in-chamber volume, displacement, and volume ratios in order to predict performance is harder than in a reciprocating piston device as the geometry is not as simple. However, there are multiple published papers concerning this [68, 72, 73] and once the equations are known it is relatively easy to incorporate them into numerical models.

2.7.6. Advantages

Many literature sources bring attention to the different advantages that Wankel devices have. Weerasinghe and Hounsham [74] state that a Wankel expander has some of the advantages of both the reciprocating piston and turbine expanders and that it performs better than piston expanders. One of the Wankel expander's main advantages is that it has few moving parts, making it reliable and durable [53], not to mention easy to assemble. Wu et al. [75] pointed out that Wankel devices have a simple structure, this makes them easy to manufacture and assemble. A Wankel compressor does not require inlet valves, further reducing the part count, causing less flow restrictions and again increasing durability [70]. Similarly, a Wankel expander does not require outlet valves and benefits in the same way.

Another prominent advantage of Wankel devices is their low noise and vibrations during operation [53, 68, 70, 75, 76], which is derived from their ease in balancing. Being well balanced contributes to their ability to reach high operating speeds [68, 76] and the fact they have a higher maximum speed means they have a large operating range [53, 77]. Ogura [70] also suggests their operating speed lends them well to be connected directly to a generator.

Due to the way Wankel devices operate, they have twice the power pulses per revolution compared to reciprocating piston devices. This leads Wankel engines to have twice the power of piston engines [68] and therefore, a higher power to weight ratio [78]. A combination of twice the power pulses per revolution and a much higher maximum speed mean that Wankel devices have much smoother operation [70].

The previous points, plus their ability to operate with low flow rates, their small and lightweight construction and their good efficiency, means they are well suited to small-scale applications [75]. Finally, Wankel expanders can handle two-phase flows [53, 76], increasing their durability and reliability further and broadening their areas of application.

2.7.7. Disadvantages – Wankel engines

The focus of the Wankel machines' development has been on Internal Combustion Engines (ICEs), in this configuration its largest disadvantage is the delayed combustion characteristics [79]. This is due to the shape being

inefficient for combustion, the large surface causes extra heat loss during combustion and the long thin profile of the combustion chamber hinders flame propagation, both leading to increased emissions and reduced efficiency [80]. Furthermore, as there is no inherent lubrication system, oil is burnt with the fuel further increasing emissions [80]. All of this leads to the fact that the fuel economy and emissions are worse than reciprocating and gas turbine engines [68].

Another disadvantage that many sources speak of is the sealing of the expander, specifically the sealing and lubrication at the rotor apexes and corners. Becker [80] points out that the sealing is required in both the radial and axial directions, which will each see different conditions. The development of the combustion engine has provided a lot of research and improvements and [53] claims the lubrication problems have now been solved, but a lot of the research is kept as company secrets.

Another mentioned disadvantage when compared to reciprocating piston engines is the peak torque is less, however the power output is much higher [70]. Therefore, this is less a disadvantage and more of an indication that Wankel machines are more suited for less torque important applications i.e. where acceleration is not as important.

2.7.8. Disadvantages – Wankel Expander/Compressor

The major disadvantage for the engine would therefore not affect an expander/compressor as there is no combustion. However, Wankel

compressors/expanders require valves for efficient operation, whereas Wankel combustion engines do not [81]. Antonelli et al. [82] stated that admission grade (inlet cut-off ratio) is important to avoid over/under expansion in an expander. The outlet valve is required on a compressor in order to separate the compression process from the discharge process. This prevents backflow when the pressure in the compressor is less than the outlet pressure.

Some research and development of compressors and expanders without valves have been done. Zhong et al. [83] developed a Wankel compressor for a small-scale refrigeration device. This compressor had no inlet or outlet valves, however no mention of the efficiency of the compressor was made in this paper. Wu et al. [75, 84] also developed a compressor without valves, as it was said that valves were not feasible due to the small size. Both these cases prove that compressors can work without outlet valves, however as neither make mention of the efficiency, it was probably not an important parameter in these projects and assumed to be low.

Weerasinghe and Hounsham [74] studied a Wankel expander without valves. In this case the efficiency is not mentioned again, this is possibly because power to weight along with simplicity is more important, due to the very limited space in the application.

Badr et al. [69] mentions that Wankel expanders can be run either with or without valves. In a subsequent paper they state that Wankel engine designs

can be used as expansion machines, but the main design changes are with the breathing system, which could include valves [85]. It is also mentioned that the location of the inlet ports on the periphery is advantageous, as there is less flow resistance and higher dynamic pressure on the rotor. However, this results in efficiency loss as the apex seal passes the port, briefly creating a leakage path each time [86]. The outlet ports don't require valves if they are located on the side of the expander, provided they are designed to open at maximum volume and close at minimum volume [85]. Finally, it is noted that the outlet ports should be large enough to limit the flow Mach number to 0.25 [85].

Antonelli et al. published a series of papers on their development of a Wankel expander with inlet and outlet valves, with tests of air, refrigerant and steam as working fluids. Their developed prototype is shown in Figure 2.19, where the valves are driven by two belts. They mention that the use of valves makes the Wankel expander more complicated than the scroll, however it is more suited to higher electrical output [76]. The main problem they encountered was pressure losses over the valves at high speeds, with the experimental results only matching the ideal cycle at 100rpm and not higher [77]. They go on to say that the pressure losses over the valves affect the admission amount, which in turn reduces the efficiency [82].

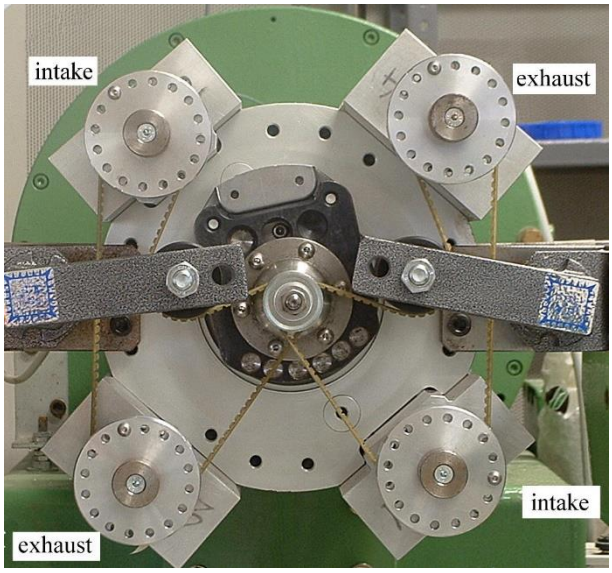


Figure 2.19 – Wankel expander with valves designed by Antonelli et al. [77]

Antonelli et al. go on to analyse and improve the valves through CFD simulations, studying their discharge coefficients, finding a better designed valve much improved the efficiency [86], this suggests valves are critical in applications where efficiency is important. Finally, they mention that because the Wankel expander runs faster than the reciprocating piston, the pressure drops over the valves are always going to be larger, due to higher flow rates. However, the higher speeds reduce losses due to leakage and heat exchange as there is less time for them to occur [55, 87].

Zhang et al. [88] looked at improving the volumetric efficiency of a Wankel compressor by utilising the residual high-pressure gas in the clearance volume. A technique called 'overflow' was used, where the residual high-pressure gas is passed to the low-pressure chamber that is being compressed, thereby boosting the low-pressure. This technique improved volumetric efficiency by 16% and increased the cooling capacity of the attached refrigeration system by 26%.

2.7.9. Sealing and Lubrication

A complete sealing system is shown for a Wankel expander in Figure 2.20 [89]. As previously mentioned, sealing and lubrication is often thought to be a large problem for Wankel machines. However, much research has been accomplished in this field for Wankel ICEs and with the compressor and expander variants. In 1991, Badr et al. [53] said that the lubrications problems for ICEs have now been solved. Therefore, it can be assumed that further improvement has made it less of an issue. The main issue faced by them was the emulsion of the oil with the steam/water at the outlet, which was hard to separate. However, for compressed air applications, oil separation technology used for screw expanders achieves 99.7% oil removal [85]. Other options are available with solid lubrication technology (PTFE/graphite), where the parts can be coated with or made from a low friction material. Unfortunately, the solid lubricant does wear away, resulting in regular maintenance [85].

Figure 2.20 - Seals for a Wankel expander prototype [89]

The lubricant and working fluid can be mixed as a miscible solution [70, 75, 84], if the Wankel device is part of a closed cycle such as in refrigeration. This removes the need for oil pumps, separators and tank increasing the simplicity of the device and reducing its size and ancillary power losses.

2.7.10. Apex Seals

Zhang et al. [90] calculated that 13.55% of a Wankel compressor's friction is from apex seals and 63.29% from the rotor sides, crankshaft sides and housing sides. However, Zhang and Wang [91] claimed that 38% of friction comes from apex seals and 40% is attributed to side the side plate. Weston [68] claimed the friction contribution of apex seals is estimated to be 25%, therefore there is a lot of ambiguity in this area.

The reduction in apex seal thickness reduces the friction contribution [68]. Furthermore, double layered leaf springs can be used to reduce the width of the spring to allow smaller seal widths. Using specially derived equations to

design shape conforming apex seals for the housing can improve sealing effectiveness, with thicker apex seals allow higher conformity [92]. Therefore, there is a trade-off in this between better sealing and lower friction. Multiple apex seals are recommended instead of one wide apex seal, this may have some of the benefits of both thin seals and the better conformity. It is stated that 80% of the forces on apex seals come from the gas pressure forces, this suggests that speed would not have a large effect on the friction due to apex seals.

Various computer models and simulations have been created to analyse the apex seals. Pennock and Beard [93] calculated that the friction between the apex seal and its rotor groove when moving in and out is insignificant. Furthermore, the speed fluctuations caused by power pulses was found to have negligible effect on the seals' performance. Zhang et al. [94] analysed the apex seal springs, observing the spring force has a negligible contribution to sealing effectiveness, except at start up. Being able to eliminate these considerations when designing an apex seal makes the process much simpler.

Picard et al. [95] found that, at low speeds side, corner and apex seals all had about an equal share of the leakage. However, as the speed increased, the contribution of the corner seals leakage reduced. Phung et al. [96] created a useful numerical model that predicted apex seal movement, vibration and lift off. This is very useful to predict the leakage for a designed system, however results using the model were only given for model validation purposes and no new insights were given.

Coating the housing surface in different material has been suggested to reduce the friction of the apex seals. Treated alloy-coated housings have been used successfully in Wankel combustion engines in an attempt to reduce friction [68]. Use of a diamond-like-carbon coating for the housing surface provided a 50% decrease in power loss at 7000rpm and 54% at 17000rpm [97]. Further research into housing and seal materials and coatings could significantly improve the performance of Wankel devices.

2.7.11. Oil Film

An important feature for the seals of any dynamic machine is the hydrodynamic oil film formed between the seal and the sealing surface. This film helps in sealing, reduces wear, and reduces friction. Therefore, understanding the oil film and its dynamic behaviour is very helpful. Wankel machines provide a distinct advantage to the oil film formation, as the seals are always rotating in the same direction [78]. If the seal stops and changes direction, as in a reciprocating piston machine, the oil film is more likely to break down.

Zhang et al. [98] performed a numerical investigation of the oil film at Wankel apex seals. It was found that the oil film thickness increased approximately linearly with speed. Additionally, at a low speed the oil film is usually thinner and there is a larger deviation between the thinnest and thickest points in a revolution, meaning increased friction when the apex seal travels further in and out of its slot. This leads to increased wear on the side of the apex seal

and the rotor groove, which in turn leads to other problems with the seal, such as excessive rocking. Less apex seals friction and wear should occur at high speeds, provided sufficient lubrication is available to the seal. Furthermore, certain crank angles where the oil film was thin were found to coincide with 'chatter' marks that were found in experimental tests. Knowing where the problems with chatter will occur is very useful and allows investigation into why this phenomenon occurs and ultimately in how it can be prevented.

Francesconi and Antonelli [89] suggests that at low pressures and temperatures there is no need for a lubrication film at all and that innovative materials for the seals or housing can be used instead. However, this would require further research into the complete effects a lack of lubrication would cause.

2.7.12. Seal-Less Wankel Machines

For various reasons, such as increasing simplicity and lack of space, some Wankel machines have been developed without certain seals. Robertson and Woigemuth [99] mentioned that in some vane expanders, which did not have side seals, increasing the side clearance from 25μm to 50μm caused an increase in efficiency loss of between 25-35%. This suggests the gap size in similar Wankel machines is very important as well. Heppner et al. [100] developed a Wankel compressor with no side seals. They state that if a machining tolerance of 6μm is achieved then the desired compression ratio can be attained. Wu et al. [75] also developed a small-scale compressor

without side seals, they drilled holes through the sides of the rotor, allowing the pressure on both sides to equalise, preventing the rotor from being pushed one way and contacting the side housing plates.

Garside [101] Developed a Wankel compressor without side seals. It is stated that oil flooding helps prevent leakage through the side gaps, as oil is supplied to the rotor central area, producing a positive pressure gradient across the side clearances into the chambers. It is stated that side clearance of 0.05mm is possible with current machining technology, they go on to state that this size gap only provides viable performance with oil flooding.

Wappenschmidt et al. [102] developed a Wankel blood pump, which had no side seals or apex seals. It was noted that the highest fluid velocities were through the apex gaps. Although this project proved the feasibility of a Wankel pump without apex seals, it may not be feasible as a compressor or expander with gas and does demonstrate that leakage through the apex gaps will be larger than leakage through side axial gaps.

2.7.13. Wankel Gears

Wankel machines are generally designed with an internal gear on the rotor and an external gear on the housing, as shown in Figure 2.21. The gears ensure that the rotor turns at a third of the speed of the shaft. This keeps the rotor moving correctly and prevents it colliding with the housing. Multiple sources have stated that the purpose of the gears is not to transmit power, but only guide the rotor [68, 70]. This makes designing the gears much

simpler as torque, stress and the resultant wear are not important factors. This is very beneficial as there are multiple other geometrical constraints on the gears including centre distance, gear ratio, diameters, and widths, that all need to be designed for specific machines.

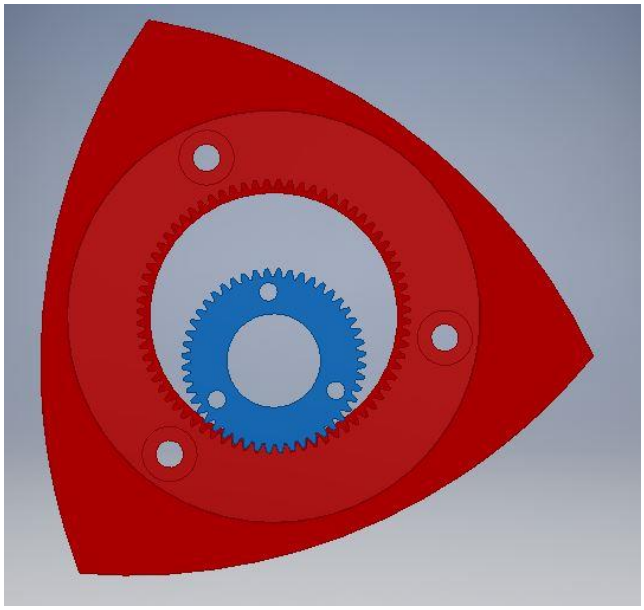


Figure 2.21 – Wankel expander gears with rotor

Wan et al. [103] designed a Wankel like pump, which did not require gears, instead using a pin in a groove with cam movement. This design is simpler than the gear type and as such it can be used for much smaller scale devices. However, this is not technically a Wankel device and as such a lot of the other research may not hold true for this device. This device may be worth further consideration but needs more research.

2.7.14. Materials

Not much research has been published on the materials for specifically Wankel compressors and expanders and although a lot of development has

been done on the combustion engine, much of that is unpublished. However, Heppner et al. [100] observed that a Wankel compressor could be manufactured from a variety of materials as there are less harsh temperatures and much lower pressures compared to combustion engines. This would also be the case with expanders.

Furthermore, research has been started on changing the materials for the Dearman engine discussed earlier in Section 2.3.1 It looked at replacing the traditionally metal components with polymers (that are lighter and provide lower friction) as temperatures are much lower than traditional combustion engines [104]. The research in this area could be directly applied to a Wankel expander's materials.

2.7.15. Small-Scale Considerations

Wan et al. [103] stated that in smaller scale devices, viscous forces dominate, which results in majority of laminar flow. However, this would not be the case if the device were scaled up, as due to the surface area-volume ratio, the surface tension and diffusion does not scale linearly with size. Therefore, results from a small-scale test device would not be representative of a large-scale version.

Zhang and Wang [88, 105] Suggests that the higher surface area to volume creates problems for lubrication of small devices. Another consideration is the space for important components, as Wu et al. [75, 84] could not find space to

include outlet valves on their small-scale compressor, and therefore designed it without.

2.7.16. Simulation Considerations

Wappenschmidt et al. [102] and Wan et al. [103] performed CFD on Wankel (or Wankel-like) machines, using the immersed solid technique to generate rotor wall movement. In the immersed solid technique two meshes are produced, one for the fluid domain trapped by the housing, and one for the solid rotor. The method utilised in their research was 'direct forcing method', whereby the cells of the fluid mesh which overlap with the rotor geometry are forced to travel with the velocity of that solid, therefore at rotor-fluid interface, the fluid behaves similar to if there was a moving wall there. The one downside to this method is that it becomes more complicated to accurately model wall conditions such as wall slip, boundary layers and heat transfer.

For Wankel machine simulations, another disadvantage of the immersed solid technique, when modelling apex gaps, is the need for high density mesh around the whole housing periphery. Alternatively, using the dynamic mesh method, where there is a single mesh in which the walls move and the interior mesh quality is maintained using smoothing and remeshing techniques, only the mesh at the apexes would need high density. Therefore, the immersed solid technique requires a mesh with a large number of cells, leading to a dramatic increase to computer processing power and time (44 hours for a simulation [102]).

Advantages to the Immersed solid method include less computational effort on updating the mesh, less instability as the mesh is kept at a constant quality and less possibility of interpolation losses occurring when the mesh is changed [102]. Therefore, the benefits and drawbacks of the different methods should be carefully analysed and compared, considering the overall computational time and accuracy of the results.

Wappenschmidt et al. [102] recommended using pressure boundary conditions instead of flow rate or velocity when simulating the Wankel devices. This is to aid simulation stability, with the flow rate (or velocity) boundary conditions, the value needs to be precisely controlled with reference to crank angle, otherwise the simulation fails. Finally, it was noted when comparing their CFD to experiments, which utilised particle image velocimetry (PIV) to observe the flow, that the CFD accurately predicted the flow velocities in many places and that CFD was more useful than PIV as it could 'see' more areas.

Zhang and Wang [88, 105] developed 2D CFD simulations on a Wankel compressor using Ansys Fluent with dynamic mesh controlled by a UDF for the rotor. In the simulation, a $5x10^{-6}$s timestep size (6667 timesteps per revolution) was used, this is a very low timestep and is required more for the dynamic mesh than the stability of the flow solver. In this CFD model, the boundary walls were all considered adiabatic, no wall friction was considered and no leakage between gaps or seals. The results had a 16% deviation between theoretical calculations and CFD, however no mention to how either

compares to experiments. The mesh was refined to always have a minimum of 3 layers of cells between walls, as this is considered the minimum for fluid flow calculations. This is because the 2 cell layers adjacent to either wall contain the flow with wall effects, leaving the central layer to model the remaining flow.

2.7.17. Experimental Considerations

This section looks at experimental techniques and considerations that were reported in the literature, that can be used when designing experimental processes from a Wankel machine.

The first consideration is about in-chamber pressure and temperature measurement. The ideal placement of such sensors would be on a rotor face, as they are the only parts that are in constant contact with a single chamber's fluid. However, as the rotor rotates in all Wankel machines, this presents a problem. Usually, the solution is placing sensors in two locations around the outer housing, allowing different parts of the cycle to be captured separately and then combined when processing the results, creating a view of the whole cycle [106]. However, if a specific location on the rotor needs to be measured, such as at the seals, then there is no other option but to have the sensors on the rotor. For this, Matsuura and Terasaki [107, 108] developed a method using a slip ring with multiple channels in order to transmit the required electrical signals from the rotor to a stationary data logger. This could also be

useful if the housing were rotating as well, such as in the DKM Wankel machine.

Another technique that is mentioned is using a damping volume before the flow meter. This is to reduce the pulsating effect produced by the Wankel expander [82]. Finally, Antonelli et al. [77] explains the best practice for finding the isentropic efficiency of the experimental expansion device. This uses the measured power divided by the ideal isentropic power for the same mass flow rate.

2.7.18. Performance of Machines in Literature

The performance of Wankel machines is quoted in multiple published works, these are useful to understand the current state of the technology and later in this research, as comparison values.

Wu et al. [75, 84] performed experimental tests on a miniature Wankel compressor, reporting a maximum value of 70% for volumetric efficiency, a compression ratio of 3 and shaft power of 95W. The power is expected to be low as the compressor only had a displacement of 3.5cc and high efficiency is much harder at these small sizes. Isentropic efficiency is not mentioned.

Garside [101] reported his Wankel compressor should be able to achieve isentropic efficiencies of 75% for a 7.5kW device, but no experimental results are mentioned.

Badr et al. [53] analysed the performance of a Wankel expander, predicting an isentropic efficiency value of 63%, but the highest experimental value achieved was 21%.

Various papers reported isentropic efficiency values of between 86% and 90% and volumetric efficiency of 85%-90% for various numerical models of a Wankel expander, with a maximum power output of 21kW [77, 86, 89, 106]. They mention that a compromise must be made between power and efficiency, as if the expander is optimised for efficiency, it loses out on power and vice versa. Experiments were successfully performed, but the efficiency values are not mentioned. They also pointed out that ancillary friction, bearing friction and seal friction are all highly dependent on speed, but not much on the pressure used.

Finally, Weerasinghe and Hounsham [74] performed CFD simulations on a small-scale Wankel expander without valves and reported that it could achieve 35W, corresponding to a specific power of 1400W/kg, taking the weight of the expander into account. However, there are no efficiency values reported.

2.8. Chapter Summary

- Global energy supply will benefit from new energy storage methods to help the utilisation of renewable energy sources.

- LAES is a promising system for low-cost energy storage integrates with multiple other thermodynamic systems to create further efficiency increases.
- Small-scale gas liquefaction will benefit both the LAES and many cryogenic systems helping to decentralise energy production and remove associated costs and losses.
- Small-scale liquefaction systems available are inefficient and have not received much development.
- ORC is another promising system that can make use of low-grade waste heat.
- The development of low-cost high-efficiency expansion devices are very important to both LAES and ORC systems.
- Different gas expansion devices have many advantages and disadvantages over each other, with none being best for all applications.
- The Wankel expander has a simple construction, low noise and vibrations, a high power to weight ratio and is reliable and durable making it very attractive for small scale applications.
- Standard Wankel expanders require added external valves to achieve their maximum efficiency.
- Wankel expanders with seals require in-chamber lubrication, but there is much past development in this area for the Wankel combustion engine.

- To understand how the performance of liquefaction systems is impacted by expander efficiency, in comparison to other components, cycle simulations were developed and utilised, as shown in Chapter 3. The development of a Wankel expander for use in liquefaction and/or ORC systems is then described in Chapters 4 to 7.

CHAPTER 3

GAS LIQUEFACTION CYCLE MODELLING

3.1. Chapter Introduction

Small-scale liquefaction is useful for various applications, including LAES systems, cryogenic cooling and on-site liquid air/nitrogen/oxygen production for medical or industrial uses. However, the performances of the cycles need to be analysed in terms of operating parameters and individual component efficiencies, as both may significantly change in small-scale system. To quickly and efficiently perform this analysis, MATLAB numerical models have been produced for a selection of current and new gas liquefaction cycles.

3.2. The Gas Liquefaction Cycles Studied

All six cycles analysed in this chapter were based on the Linde-Hampson cycle. The Linde-Hampson cycle is the simplest of all gas liquefaction cycles, requiring only an air compressor, a cooler, a heat exchanger, an expansion valve and a liquid/gas separator. A diagram of this cycle is shown in Figure 3.1. The gas flows in a cycle, being compressed and cooled with ambient temperature liquid, then undergoes further cooling from the returning gas. Next the gas is allowed to expand through an isenthalpic expansion valve back to low pressure. In doing so, the gas is further cooled. If any part of the gas has liquefied at this stage it is removed and stored by the separator, otherwise it returns through the heat exchanger, cooling the incoming higher-

pressure gas. Theoretically it should exit the heat exchanger back near ambient temperature and pressure, therefore it should not matter if the gas is recycled or new gas is used. This trait is the same with all the gas liquefaction cycles studied in this section.

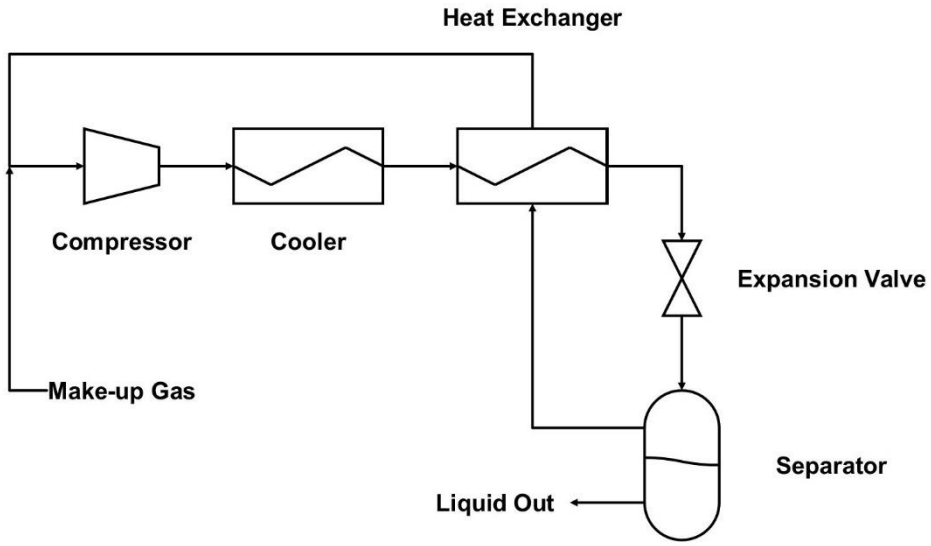


Figure 3.1 - Linde-Hampson Cycle

The next cycle analysed is called the Claude cycle. This cycle provides a modification to the previously mentioned Linde-Hampson cycle, in order to drastically improve efficiency. This modification includes the addition of two further heat exchangers and an expander. Furthermore, it will be required that there is some two-way valve that can be used to split the flow at a given fractional amount.

The idea is that the expander will provide the majority of the cooling effort, as it will ideally be operating close to the isentropic conditions. The isenthalpic expansion valve is only utilised as most expanders cannot tolerate the formation of liquid in them.

The Claude cycle is shown in Figure 3.2. The first heat exchanger performs similarly to the one in the Linde-Hampson cycle, being that it cools air coming from the compressor/cooler with the outgoing air, about to be recycled. After the first heat exchanger, the incoming air is split into two streams, one passes to the expander and the other continues to the second heat exchanger. The second heat exchanger is where most of the heat transfer occurs, as it has a mixture of streams straight from the expander and the from the expansion valve entering its cold side. The final heat exchanger simply gives a small amount of cooling to the gas just before the isenthalpic expansion, to try and give the highest yield possible.

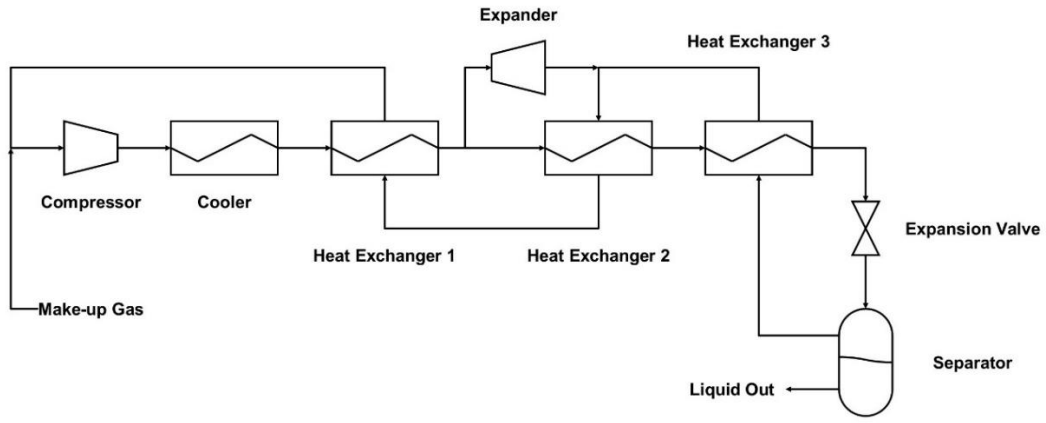


Figure 3.2 - Claude Cycle

The next two cycles analysed are the Kapitza and the Heylandt, cycles shown in Figures 3.3 and 3.4. Both are very similar to each other and are very simple modifications on the Claude cycle, with both removing one heat exchanger from it. For the Kapitza cycle, the last heat exchanger is removed. Therefore, the gas that passes through the expander is directly mixed with the gas returning from the expansion valve/separator. Although, this cycle may result

in a slight reduction of efficiency, due to a lower yield. It becomes a simpler and smaller system to implement which could be more desirable.

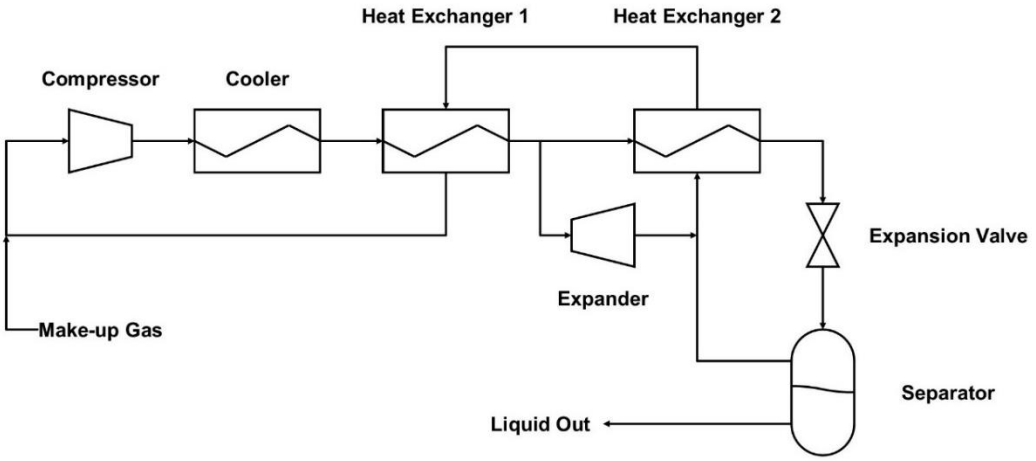


Figure 3.3 - Kapitza Cycle

The Heylandt cycle does the same thing with the first heat exchanger. With the idea being the same. This cycle reportedly only produces good efficiency at higher operating pressures. This is because the gas going through the expander receives no pre-cooling and the cooling effect due to pressure-drop is much less when starting at higher temperatures.

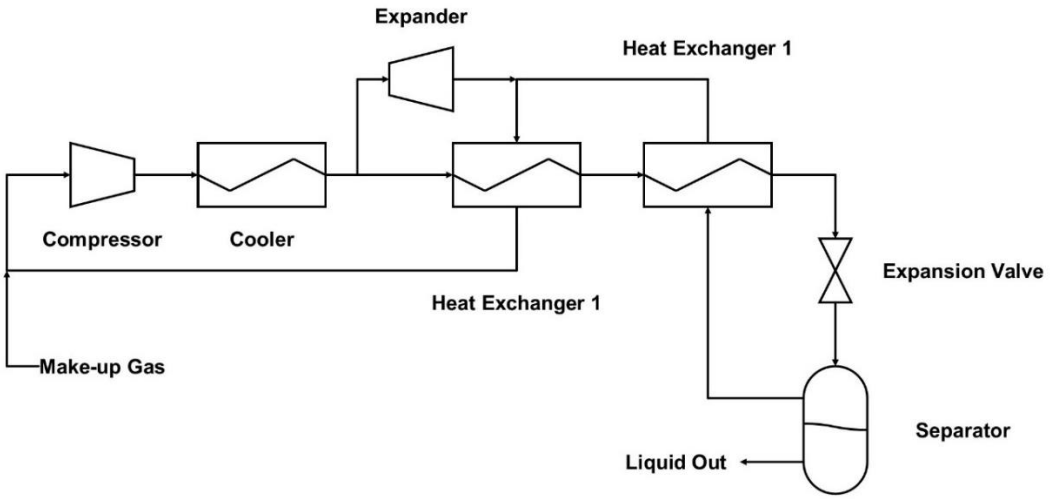


Figure 3.4 - Heylandt Cycle

Aside from these 4 cycles that have been used before, 3 new cycles with modifications to these are analysed, to assess the possibility of further efficiency improvements. These cycles will be named New Cycle 1, 2 and 3 for simplicity.

New Cycle 1, shown in Figure 3.5, took the Kapitza cycle and modified it so that a fraction of the liquified gas yield was rerouted back through a heat exchanger. This further cooled the incoming gas just before the expansion valve. The extra cooling will increase the yield, but the cycle needs to be numerically analysed to understand if the increase in overall yield can offset the amount removed from it.

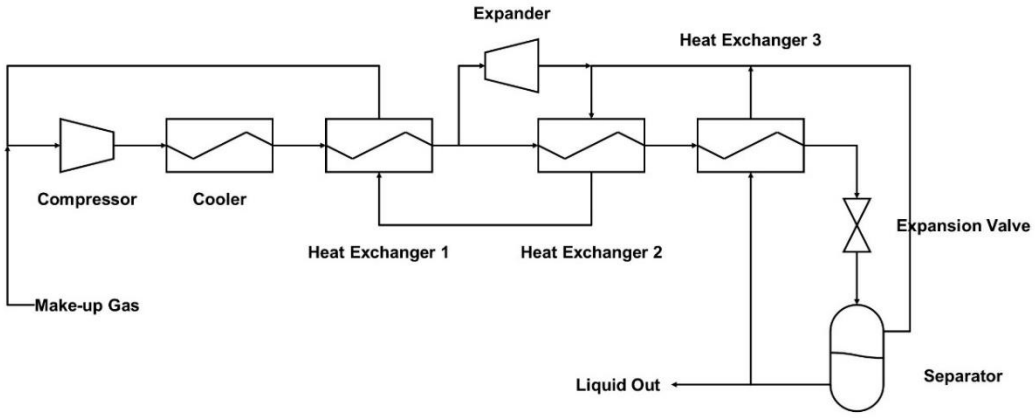


Figure 3.5 - New Cycle 1

New Cycle 2, shown in Figure 3.6, modified the previous cycle (New Cycle 1) by removing the heat exchanger that the returned liquid passes through and instead mixing it directly with the returning gas. This may not provide as low temperatures just before the expansion valve. However, one less heat exchanger would be highly beneficial, especially for a small-scale system.

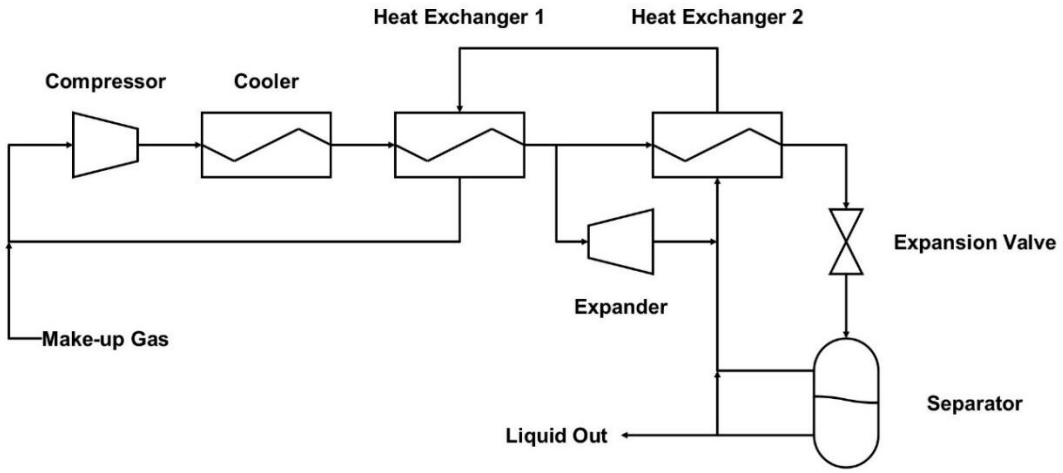


Figure 3.6 - New Cycle 2

New Cycle 3, shown in Figure 3.7, took the Claude cycle as a base and added a second expansion device to it. This should allow the incoming gas to receive two stages of cooling from expansion devices, with one expansion device effectively precooling the gas before the second expansion device. Again, analysis is required to find out if the yield increase from this would offset the fact that the incoming gas now has two streams split off it, reducing the amount through the expansion valve and therefore the amount contributing to the final liquid yield.

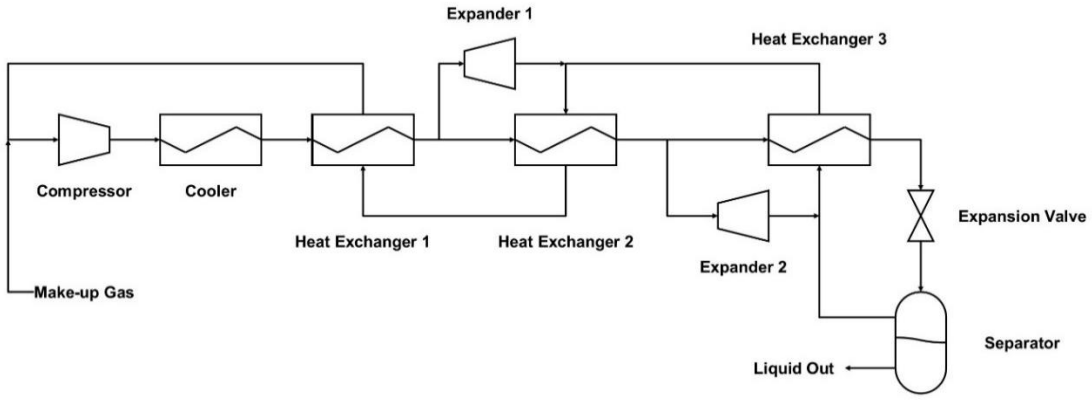


Figure 3.7 - New Cycle 3

3.3. Components used in the Cycles

3.3.1. Compressor

The first component used in all the cycles are gas compressors. These compressors can be of many varieties, with different performance characteristics as studied in the earlier literature review chapter. They may be single stage or multiple stage depending on their compression capabilities, however within these cycle analyses the only characteristics which are considered are the overall compression ratio, the flow rate and the isentropic efficiency. These three factors are used in a 'black box' concept, where inside there could be any type and any number of compressors.

3.3.2. Expansion Valve

The next component that all cycles utilise is an expansion valve. This is a very simple component, that can be realised practically in multiple different ways. The simplest example would be a restriction in the pipeline, allowing a higher pressure to be maintained on one side. However, a better option would be an adjustable valve of some description, so that the pressure-drop can be fine-tuned to the optimal level when the system is running at different operating conditions. In the cycle analyses the expansion valve can be very simply modelled. Any pressure-drop through a valve (assuming no leakage or heat transfer) can be modelled as a constant enthalpy process.

3.3.3. Gas Expander

The third component is the gas expansion device. This device provides a similar function to the expansion valve, however it uses the pressure-drop to produce work output. This work output can be put back into the system to provide power for the compressor increasing the efficiency of the system. Furthermore, as energy is being removed from the system in the form of work done, and the outlet pressure remains the same, the internal energy of the outlet gas is lower, which means a lower temperature at the outlet. As all the systems are focussed on cooling to the point of liquefaction, this extra cooling further increases efficiency. The expansion device, like the compressor before, is also treated as a 'black box' in the analyses, as there are also many types and setups available. Therefore, only the isentropic efficiency of the device needs to be known.

3.3.4. Heat Exchanger

All of the cycles use heat exchangers, and most utilise multiple. There are multiple different types of heat exchanger, with different directions of flow, different locations of the steams internally and different heat transfer materials. For all of the cycles counter current heat exchangers are used, wherein the outlet of one stream is closest to the inlet of the other stream and vice versa. This setup allows for the best heat exchange and lowest temperature outlets, which is highly desirable for these cases. Heat exchangers are the hardest to analyse, as there are two separate gas streams

flowing through each and the behaviour of each is dependent on the other. Therefore, to keep the model simple and reduce computational time, the heat exchangers were also considered 'black box' devices, where the heat exchanger 'effectiveness' is given to define its performance.

3.3.5. Phase Separator

The final component that all cycles use is the phase separator. The concept of this device is very simple, it removes any liquid phase from a fluid flow. This allows the liquified yield to be collected and the cooled gas to remain in the cycle. In practice this can be achieved in various ways, most commonly involving creating a vortex flow pattern which forces the heavier liquid phase to the outside via centrifugal action, where it is deposited on the walls and drops to a collection point. In the analyses the vapour quality of the flow coming into the phase separator is determined. The flow is then split into two outlet flows, in proportion to the vapour fraction, one being saturated liquid and the other being saturated gas.

3.4. Software used for Analyses (MATLAB/CoolProp)

The coding software MATLAB was chosen to implement the models both because it is well known to the author and it easily allows the incorporation of the CoolProp fluids library. CoolProp is an open source thermodynamic properties library that allows properties of different fluids to be produced based on given values of other properties. Generally, at least two properties need to be known to define a fluid state.

3.5. Model Assumptions

A few assumptions are made in the numerical model, these are primarily done in order to simplify the simulation and speed up processing time, but also when there is a lack of data or when the actual parameters could vary largely. In any case, as these models were made to compare different cycles, the results should not be affected overly much as all the cycles will have the same assumptions.

3.5.1. Pressure Loss

Pressure loss was not accounted for in the model's heat exchangers and phase separator components, as it would be highly dependent on both the individual type and design of component used. However, the model does specify pressure at each point in the cycle and it would therefore be relatively easy to add pressure loss in future versions, if the values became known.

3.5.2. Heat Loss

Heat loss out of the cycles was not accounted for, except in the compressor/cooler section, as this is an intrinsic part of the function. The reason for neglecting the heat loss is due to lack of data to estimate the heat loss and the complexity of modelling full component systems to try and find it this way. Furthermore, the heat loss from the system should be minimal, with suitable insulation. Typical insulation for cryogenic systems involves a mixture of glass fibre and aluminium foil, which can be easily wrapped around the various parts of the cycles. Furthermore, cryogenic insulation

techniques have been developed for many years, so have a high level of maturity.

3.5.3. Leakage

Leakage of gas out of the system was not included for simplicity, which may be present in real world systems.

3.6. Overview of the Numerical Model

The code written for the analyses of the cycles is formed in a very modular way. For example, each component has a function written for it which can be called in any of the modules whenever required. Furthermore, there are also generic functions that update a fluid's state given a known pressure and enthalpy. This therefore requires that each component function gives an output of its pressure and enthalpy. To record the model's output data, structured arrays are used for each point in the cycle. This makes the code easy to follow.

As the heat exchanger functions both have two inputs and two outputs all of which change depending on each other, the solutions at the state points need to be iteratively solved. The code needs to be iterated enough times to allow all the state points to settle on a steady state solution for the cycle. This is done with a while loop which breaks if the residuals of all the state points drop below a pre-determined value. Additionally, the loop can also be terminated if an oscillating result is detected. Figure 3.8 gives an overview of

how the model operates and the full MATLAB code is included in Appendix A.

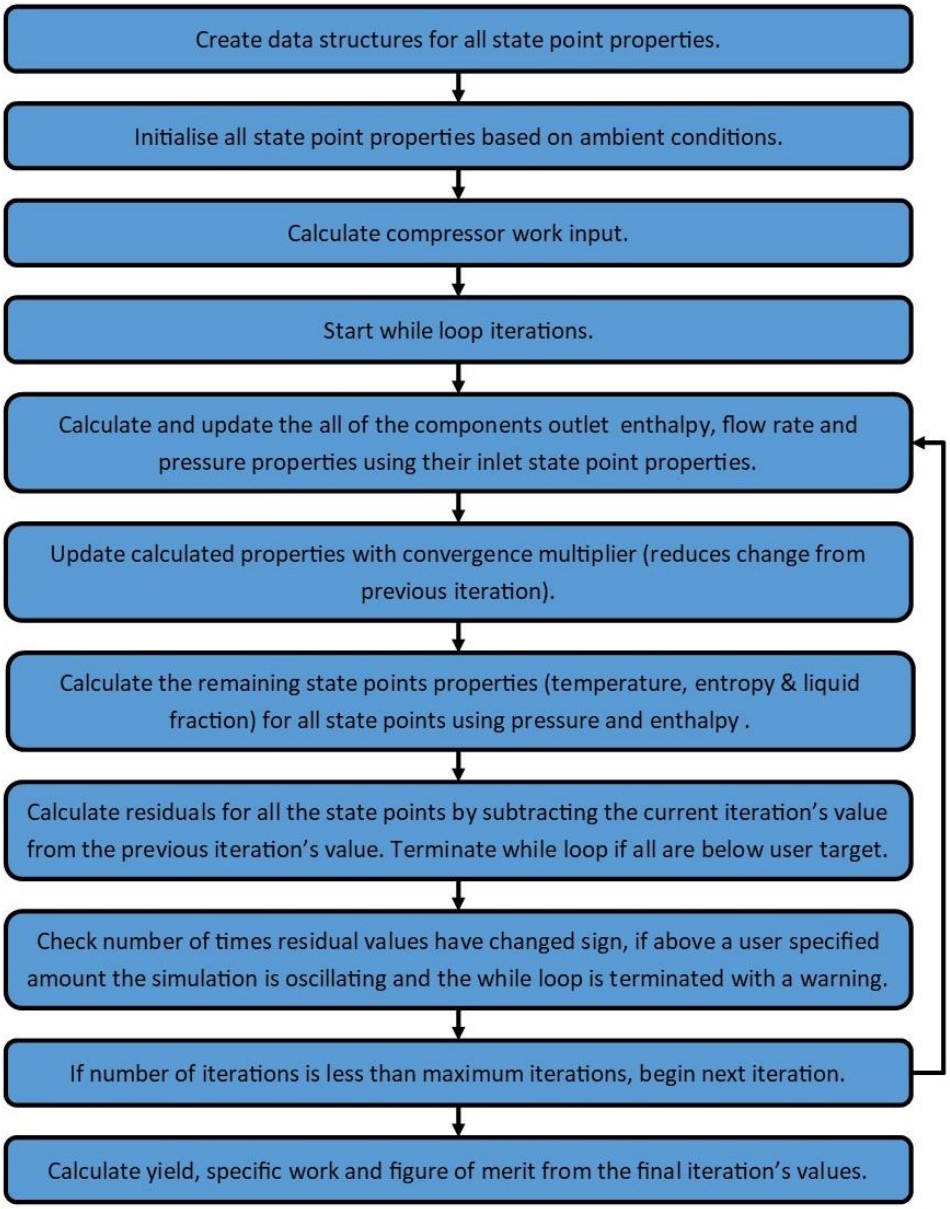


Figure 3.8 – Flow diagram for the MATLAB models

3.7. Equations used for each Component/Section

3.7.1. Compressor

As the inlet for the compressor should be near the initial ambient conditions and the output is always cooled back to ambient by the after cooler, iterative state calculations are not required for the compressor, only one calculation is required at the start. Figure 3.9 shows the inlet and outlet streams of this component, which are fully defined at the start of the model, as both are user inputs. The only calculation that is needed is the power consumption (or work input). This can be calculated from the pressure difference required, the flow rate and the overall isentropic efficiency of the compressor. Equation 3.1 describes this calculation.

$$Power_{compressor} = \dot{m}\left(\frac{h(p_{outlet}, s_{inlet}) - h(p_{inlet}, T_{inlet})}{\eta_{Compressor}}\right) \qquad (3.1)$$

Where $\dot{m}$ is the mass flow rate, h() is a function to find the specific enthalpy given two other state properties, p_{outlet} is the pressure at the compressor outlet, $s_{ambient}$ is the specific entropy, p_{inlet} and T_{inlet} are the ambient pressure and temperature and $\eta_{Compressor}$ is the compressor isentropic efficiency.

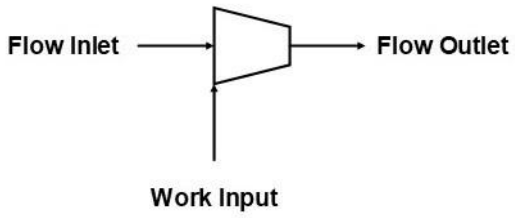


Figure 3.9 - Compressor

3.7.2. Expansion Valve

As the expansion valve operates on the theory that it is isenthalpic, it does not require any equation to itself. The expansion valve function therefore just applies the enthalpy of the inlet to the enthalpy of the outlet and sets the outlet pressure to the user desired value for the low-pressure side of the cycles.

3.7.3. Expander

The model's function for the expander component utilises the inlet pressure and temperature, the pressure-drop and the expander's isentropic efficiency. Figure 3.10 shows the expander in diagram form, it has one input to be considered and produces output data for the outlet stream and the power/work output. Two calculations need to be performed, the first to find the properties of the outlet stream and the second to find the mechanical power output. To find the outlet properties Equations 3.2 and 3.3 are used.

$$h_{outlet} = h_{inlet} - \left(\eta_{expander} . \left(h_{inlet} - h(p_{outlet}, s_{inlet})\right)\right) \quad (3.2)$$

$$Power_{expander} = \dot{m}_{inlet}(h_{inlet} - h_{outlet}) \quad (3.3)$$

Where h_{outlet} is the specific enthalpy at the outlet, h_{inlet} is the specific enthalpy at the inlet, $\eta_{expander}$ is the isentropic efficiency of the expander, p_{outlet} is the pressure at the outlet of the expander, s_{inlet} is the specific entropy

at the inlet to the expander, $\dot{m}_{inlet}$ is the mass flow rate through the expander and $Power_{expander}$ is the power output of the expander.

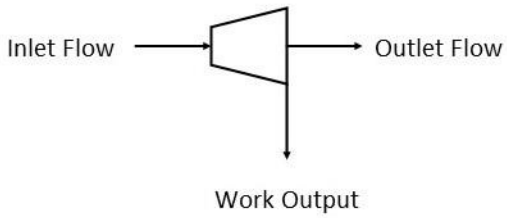


Figure 3.10 - Expander

3.7.4. Heat Exchanger

Figure 3.11 shows a diagram of the heat exchangers used in the models, the functions for these heat exchangers required multiple different steps to find the output conditions. Firstly, they need to determine the maximum possible heat transfer between the two streams and then adjust this depending on the effectiveness of the heat exchanger. The maximum possible heat transfer is found by first determining the enthalpy change that would occur to the hot stream, if it were to cool down to the temperature of the cold stream inlet, and then determining the enthalpy change that would occur to the cold stream, if it were to heat up to the temperature of the hot stream inlet. This is demonstrated by the minimum of Equations 3.4 and 3.5. This is done as these are both the physical limits on each stream. As neither limit can be exceeded, the smaller of these two enthalpy changes is chosen as the heat transfer.

Once the heat transfer is decided, it is multiplied by the heat exchanger's effectiveness and this value is used to determine the conditions of both outlets as shown in Equations 3.6 and 3.7.

Minimum of:

$$dH = H_{hot,inlet} - H_{hot,possible\ outlet\ @\ T2} \tag{3.4}$$

Or

$$dH = H_{cold,inlet} - H_{cold,possible\ outlet\ @T1} \tag{3.5}$$

$$H_{hot,outlet} = H_{hot,inlet} - EdH \tag{3.6}$$

$$H_{cold,outlet} = H_{cold,inlet} + EdH \tag{3.7}$$

Where dH is the enthalpy change from the inlets to outlets of the streams, E is the heat exchanger effectiveness, the 'hot' and 'cold' subscripts are for the hot and cold streams respectively, subscript 'inlet' refers to the enthalpy at the stream inlet and 'outlet' refers to the enthalpy at the stream outlet. Subscripts 'possible outlet @ T1' and 'possible outlet @ T2' are the theoretical enthalpies if the outlet of the respective stream was cooled to the inlet temperature of the other stream.

Aside from the above, the heat exchanger function had to be modified to account for cases where phase change or liquid phase would occur in one or both of the streams. As phase change from gas to liquid occurs at constant temperature, there can be a large enthalpy difference between liquid and gas

state for a very small temperature difference. Furthermore, because the enthalpy is calculated using the temperature, if the simulation encounters phase change, a sudden large change in enthalpy will be seen in just one iteration. This causes oscillations in the calculation of the solution and prevents the model from converging. This problem was addressed by first assessing if phase change in either stream would occur in any iteration, using Equations 3.4 to 3.7. If so, the outlet enthalpy would not be calculated as per the above equations. Instead, the outlet enthalpy would in reduced by a small increment each iteration, so that temperature was not utilised in the calculation whilst phase change conditions remained. This allowed the rest of the simulation to adjust to each new enthalpy value and prevented oscillations that would occur otherwise.

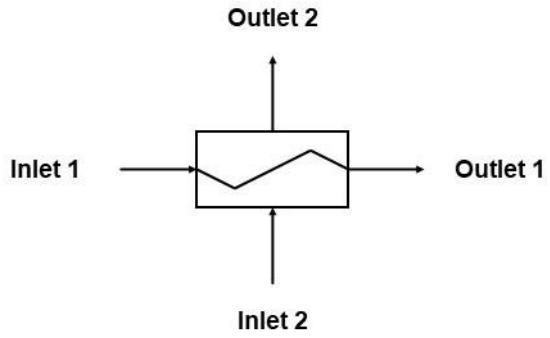


Figure 3.11 – Heat Exchanger

3.7.5. Phase Separator

The phase separator's function in the model is primarily required to determine the flow returning through the cycle, its diagram is shown in Figure 3.12. The conditions of the return stream are easily determined, if the stream is above boiling temperature, then no change is made as it is in the

saturated gas phase. If the temperature is at boiling, then the outlet stream back through the cycle is set to be at saturated gas conditions, as it is assumed that all the gas fraction of the stream is recycled, and all the liquid fraction is removed. In this case the flow rate of the recycled gas is adjusted using the vapour fraction of the inlet to the separator.

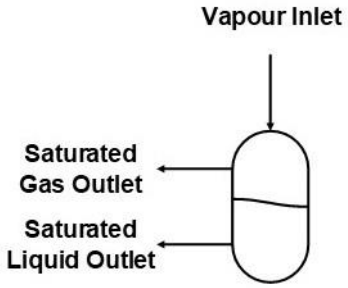


Figure 3.12 – Phase separator

3.7.6. Convergence

For any of the more complicated cycles, with many component parts, convergence on a solution is impossible using only the equations above on their own. Therefore, to allow convergence, multipliers are used. These are used when updating the mass flow rate and specific enthalpy state points in all simulations. The update process calculates the difference between the current iteration's state point value and the new value which is calculated by the equations. This difference is then reduced by the multiplier, which is set by the user and the current iteration's state point property is moved this amount towards the actual equation calculated value (essentially a form of under-relaxation). The Equations 3.8 and 3.9 describe this process.

$$h_{i+1} = h_i - \left(\frac{(h_i - h_{i+1})}{multiplier}\right) \quad (3.8)$$

$$\dot{m}_{i+1} = \dot{m}_i - \left(\frac{(\dot{m}_i - \dot{m}_{i+1})}{multiplier}\right) \quad (3.9)$$

Where h is specific enthalpy, $\dot{m}$ is mass flow rate, $i+1$ subscript is the calculated value for the next iteration, i subscript is the value for the current iteration and $multiplier$ is the user set multiplier value to control convergence.

3.7.7. Residuals

To assess how well a solution has converged residual values are used. These values are calculated from enthalpy and flow rate values at all different state points in a cycle. The residual is equal to the difference of the enthalpy or flow rate values of the current iteration and that of which it should be to satisfy the equations. Each residual value is in turn compared to a set maximum, whereby if all are lower than this, the model stops iterating and is considered converged.

3.7.8. Oscillations

Some solutions with certain operating conditions never reach convergence in the model. In this case, the model needs some method to stop, otherwise it would be stuck in an infinite loop. The first solution is to provide a maximum number of iterations. This does prevent the infinite loop problem, however,

some solutions take a large number of iterations to solve, so the maximum number of iterations needs to be fairly high.

Furthermore, some solutions start oscillating fairly early on and do not provide a fully converged solution. In these cases, it would be useful for time saving to stop the solution once oscillations are detected. Therefore, the code also counts the number of times the gradient of each state point's enthalpy and flow values change sign. This number is then checked against a user set value for the maximum number of oscillations allowed and if any value exceeds this the iterations are terminated.

At this point, that solution is marked as having oscillated, but a result is still output, given as an average of the previous x number of iterations. It was found this gave a good approximate result for comparison and was used to determine if that oscillating solution required further investigation, or if those operating conditions could be written off as undesirable.

3.7.9. Figure of Merit and Yield

Yield is the amount of gas liquefied in relation to the amount of gas passing through the compressor. This is important as it represents how fast the gas will be liquefied, if the yield is very low then a much larger system will be required for the same speed of liquefaction when compared to a higher yield system. Yield is calculated within the model's separator section, where the flow in vapour conditions is split into a saturated liquid and a saturated gas stream.

Figure of Merit (FOM) is calculated from the ideal energy required to liquefy a kg of gas and the models calculated energy required, which is derived from the compressor work input and expander work output. The FOM is essentially an efficiency value for the whole cycle in comparison to an ideal cycle. It is calculated using Equations 3.10, 3.11 and 3.12.

$$w_{ideal} = T_1(s_1 - s_f) - (h_1 - h_f) \tag{3.10}$$

Where w_{ideal} is the theoretically ideal specific work input required for liquefaction, subscript '1' is starting conditions (ambient) and subscript 'f' is saturated liquid conditions at the same pressure.

$$w_{actual} = \frac{(w_{compressor} - w_{expander})}{yield} \tag{3.11}$$

Where w_{actual} is the specific work input required for liquefaction calculated by the model, $w_{compressor}$ is the compressor's work input and $w_{expander}$ is the expander's work output.

$$FOM = \frac{w_{ideal}}{w_{actual}} \tag{3.12}$$

3.8. Parametric study

3.8.1. Optimisation

As each cycle has multiple different parameters and operating conditions that can take on varying values, an optimal result had to be found for each cycle so they could be fairly compared. This optimal result involved finding each of

the optimal operating conditions for each cycle. Therefore, a parametric study was setup for each, which in turn, altered the compressor outlet pressure and the fractions of flow diverted to the expander (in cycles with one expander). A further parametric study was setup for the cycles with two flow splitting sections, where surface plots were used to identify the optimal setups.

3.8.2. Component Efficiency and Effectiveness Analysis

The effectiveness of the heat exchangers and the efficiencies of the expander and compressor components can be set as an input for the simulations. This is done as there are huge variety of these components available in which these values vary depending on many factors. In order to find out how each of these factors affects the whole cycle performance individually, studies were setup where they were each varied throughout their ranges, allowing the individual effects to become apparent.

In all the studies, both cycle FOM and output yield are used to evaluate the performances. Table 3.1 shows the setup of the parametric study, including the values of all input variables.

Table 3.1 – Parametric study setup

Study	Cycles	X1	X2	p_c [Pa]	E	η_c	η_e
Optimisation of cycles with 2 flow diversions (high pressure)	New cycles 1,2 & 3	0.1 to 0.9	0.1 to 0.9	6,000,000	0.99	0.85	0.7
Optimisation of cycles with 2 flow diversions (low pressure)	New cycles 1,2 & 3	0.1 to 0.9	0.1 to 0.9	800,000	0.99	0.85	0.7
Optimisation of cycles with one flow diversion (high pressure)	Claude, Kapitza and Heylandt cycles	0.1 to 0.9	N/A	6,000,000	0.99	0.85	0.7
Optimisation of cycles with one flow diversion (low pressure)	Claude, Kapitza and Heylandt cycles	0.1 to 0.9	N/A	6,000,000	0.99	0.85	0.7
Optimisation of compressor pressure	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	800,000 to 6,000,000	0.99	0.85	0.7
Analysis of heat exchanger effectiveness (high pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	6,000,000	0.7 to 1	0.85	0.7
Analysis of heat exchanger effectiveness (low pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	800,000	0.7 to 1	0.85	0.7
Analysis of compressor efficiency (high pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	6,000,000	0.99	0.1 to 1	0.7
Analysis of compressor efficiency (low pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	8,00,000	0.99	0.1 to 1	0.7
Analysis of expander efficiency (high pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	6,000,000	0.99	0.85	0.1 to 1
Analysis of expander efficiency (low pressure)	Claude Kapitza Heylandt New Cycle 1 New cycle 2 New Cycle 3	0.75 0.7 0.4 0.7 0.7 0.65	N/A N/A N/A 0.1 0.1 0.5	800,000	0.99	0.85	0.1 to 1

3.9. Optimising Flow Diversion Fraction(s)

This first results section looks at optimising the flow diversion fractions. As both the amount of flow to both the expander and to the separator benefit from being increased, these parametric studies are required to find the optimum balance.

3.9.1. Comparison of All Cycles at High-Pressure Conditions

Looking at Figure 3.13, all cycles except Heylandt and New cycle 3 have a similar trend with x values. With peaks being between X = 0.6 and 0.7. The peak FOM for the Heylandt cycle occurs at a much lower x value, however the FOM value is much reduced compared to all the others. New cycle 3 shows an unusual trend, as it has two x fractions that ideally need to be optimised, not just one. This is done further down the chapter.

Therefore, of all the cycles, the Kapitza cycle would be preferable as it contains the least components and would be cheaper to produce, easier to maintain and smaller in size. The yield values of the cycles, shown in Figure 3.14, all follow the exact same trends as the FOM values, therefore FOM can be used to choose a best X values.

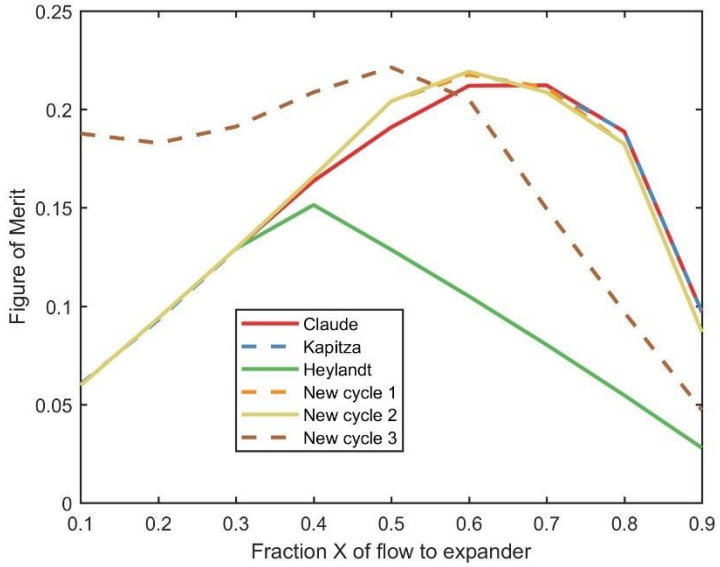


Figure 3.13 - FOM against X fraction (60bar operating pressure)

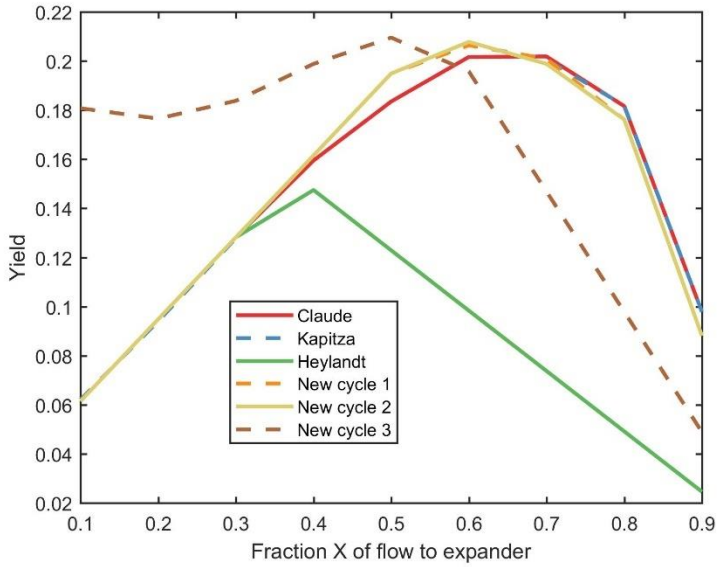


Figure 3.14 - Yield against x fraction (60bar operating pressure)

3.9.2. Comparison of All Cycles at Low-Pressure Conditions

Lower pressure results in better FOM/efficiency but a much lower yield, as shown in Figures 3.15 and 3.16. Therefore, a larger system would be required to produce the same output rate of liquid gas. The peak value for FOM occurs at a much higher flow fraction than the higher-pressure cases.

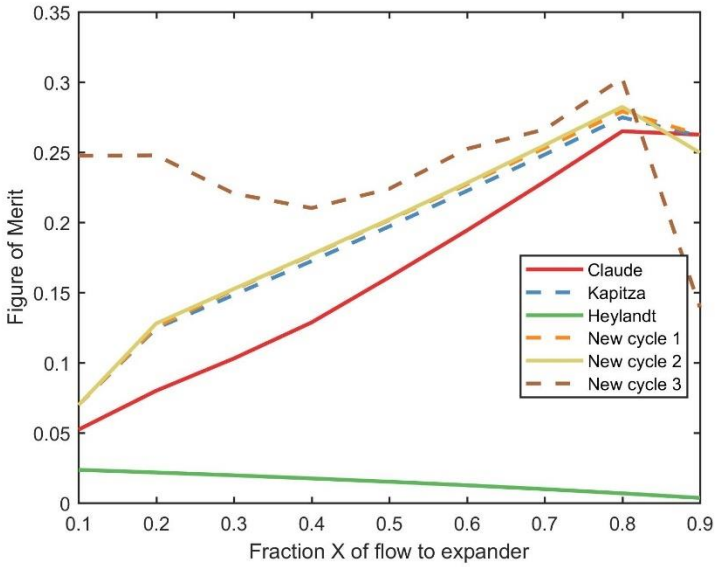


Figure 3.15 – FOM against X fraction (8bar operating pressure)

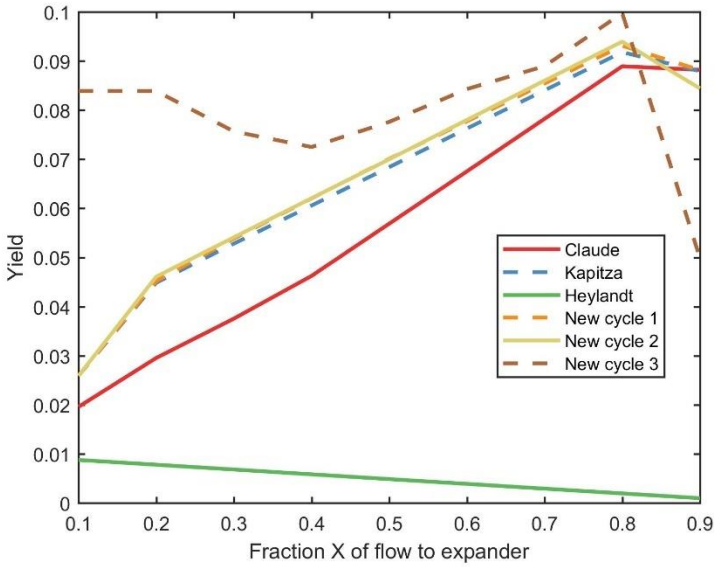


Figure 3.16 - Yield against X fraction (8bar operating pressure)

3.9.3. Optimising Cycles with Two Flow Diversions at High Pressure Conditions

New cycles 1 and 2, shown in Figures 3.17 and 3.18 respectively, both show similar trends. With the maximum FOM values lying on the X2 = 0 plane and

generally dropping off as the value is increased. As the X2 value is the fraction diverted to the new parts of the cycle, the best performance occurs when the cycles are essentially acting as the Kapitza cycle. Therefore, this modification gives no added benefit at these high-pressure conditions.

The yield at these peak FOM zones is the same for both New Cycles 1 and 2, which is similar to the less complex cycles, Therefore, there is no advantage from this point of view either.

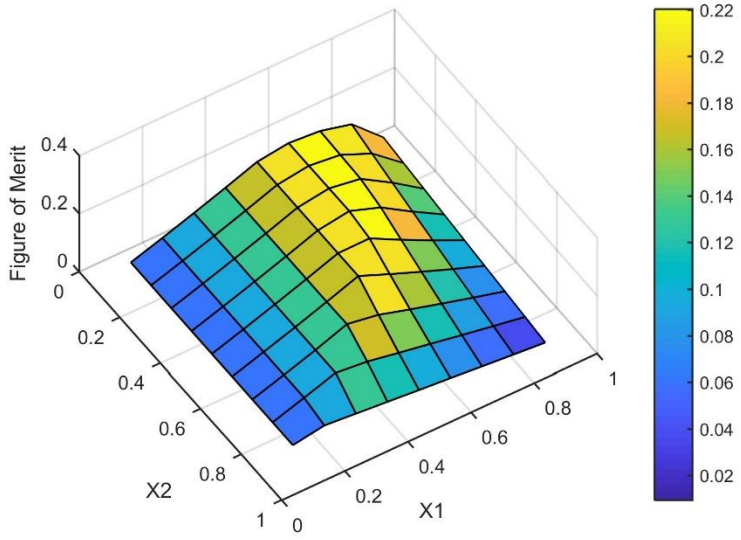


Figure 3.17 – New Cycle 1, X1 and X2 against FOM (60bar operating pressure, maximum yield = 0.2087)

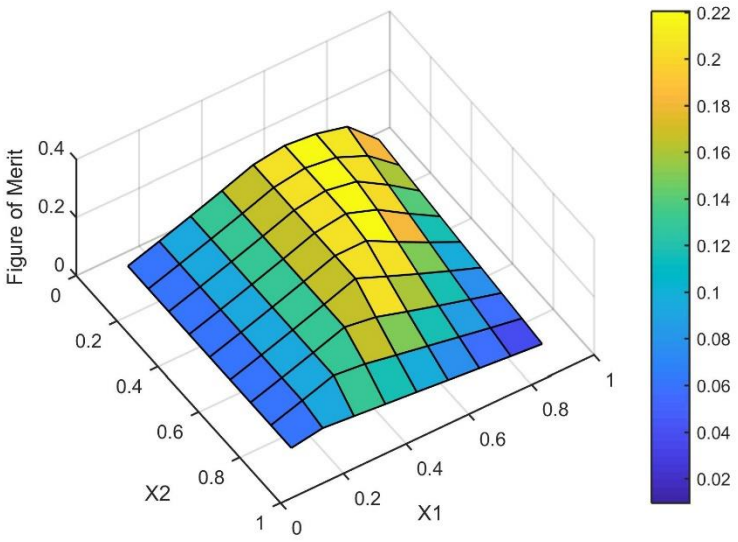


Figure 3.18 - New Cycle 2, X1 and X2 against FOM (60bar operating pressure, maximum yield = 0.2088)

Figure 3.19 shows that New Cycle 3 gives a range of peak values that span from one X=0 axis to the other. When X1=0 the first expander is not used and the cycle acts as the Kapitza cycle and when X2=0 the second expander is not used and the cycle acts as the Claude cycle. The maximum of the peaks exists where X1=X2=0.5, however this maximum is only slightly higher than the rest of the peaks, which extend to the conditions where either X1=0 or X2=0. The yield follows the same trend, with a maximum slightly outperforming all the other cycles at this pressure, but not by a significant amount. Therefore, even though there is a small improvement in FOM in using two expanders, the fact that the extra expander will increase the system's cost, size and frictional losses, it would not be worth it.

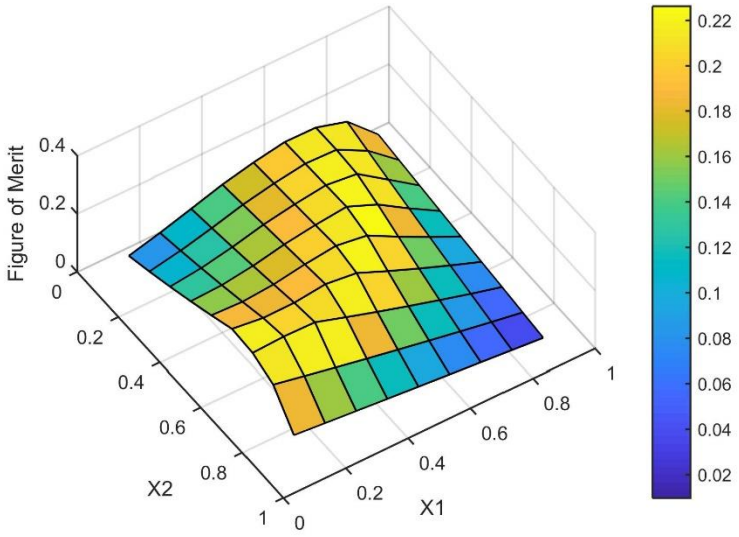


Figure 3.19 - New Cycle 3, X1 and X2 against FOM (60bar operating pressure, maximum yield = 0.2136)

3.9.4. Optimising Cycles with Two Flow Diversions at Low-Pressure Conditions

When using low pressure conditions of 8bar, the New cycles 1 and 2 provide small benefit to FOM, with New Cycle 2 performing slightly better than New cycle 1. These are shown in Figures 3.20 and 3.21. The trend remains very similar between the two cycles, however, both trends differ from their higher-pressure versions in that the peak FOM values are now located at much higher X2 values. The yield values for these two cycles are both slightly above the other regular cycles at this operating pressure. Therefore, it can be said that recycling some of the liquid yield can slightly improve the FOM if low pressure conditions are required, such as may be the case in a small-scale system.

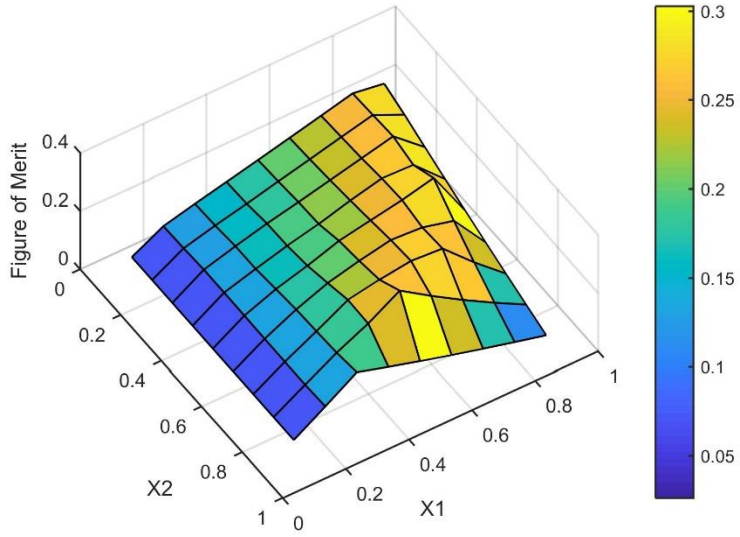


Figure 3.20 - New Cycle 1, X1 and X2 against FOM (8bar operating pressure, maximum yield = 0.0998)

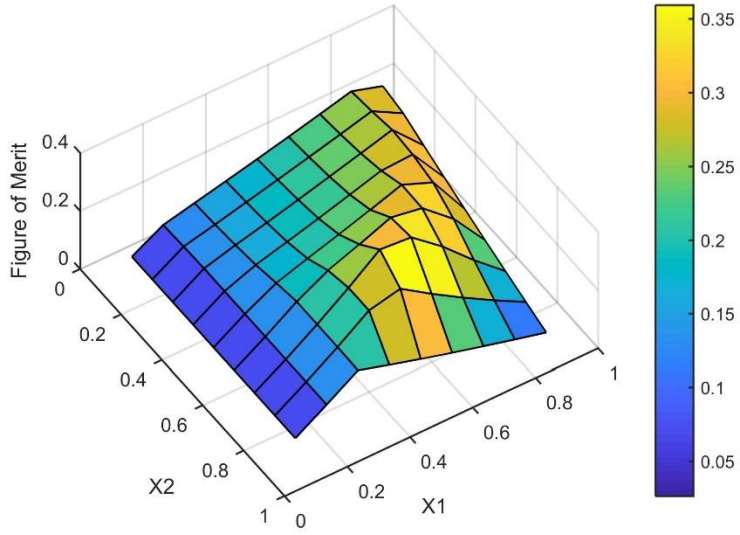


Figure 3.21 - New Cycle 2, X1 and X2 against FOM (8bar operating pressure, maximum yield = 0.1153)

When operating at low pressure conditions, New Cycle 3 has a similar trend of FOM values, this is shown in Figure 3.22. However, the curve of peaks from one zero plane to the other is shifted to slightly higher X values, with the largest FOM value at low X1 and high X2. With the low-pressure operating

conditions, it now offers significant improvement in FOM and small improvement to yield, compared to the original cycles.

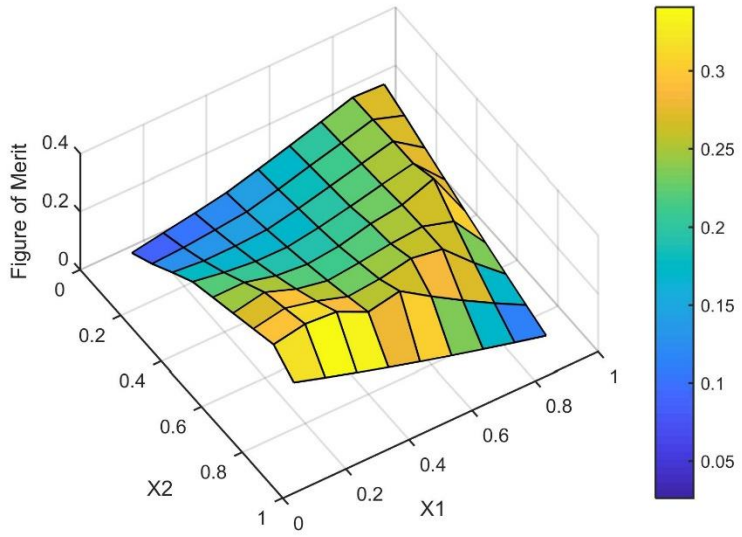


Figure 3.22 - New Cycle 3, X1 and X2 against FOM (8bar operating pressure, maximum yield = 0.1080)

3.10. Effect of the Compressor Outlet Pressure

When looking at a study where the compressor pressure is varied, there is a trade-off. The FOM and the yield for these results are given in Figures 3.23 and 3.24 respectively. There appears to be a peak in FOM at a compressor outlet pressure of 20bar, after which it steadily drops off. The yield increases at all points with increasing pressure. However, at 20bar pressure the rate it increases slows dramatically and even though the yield is larger at the high pressures, the increase after this point is relatively small. Therefore, 20bar would appear to be the optimum pressure for most of the cycles.

The Heylandt cycle's FOM and yield values both increase throughout the pressure range, unlike the other cycles. Therefore, the optimum pressure

value for the Heylandt cycle lies beyond the upper value of the simulation range, which would be too high for any small-scale system.

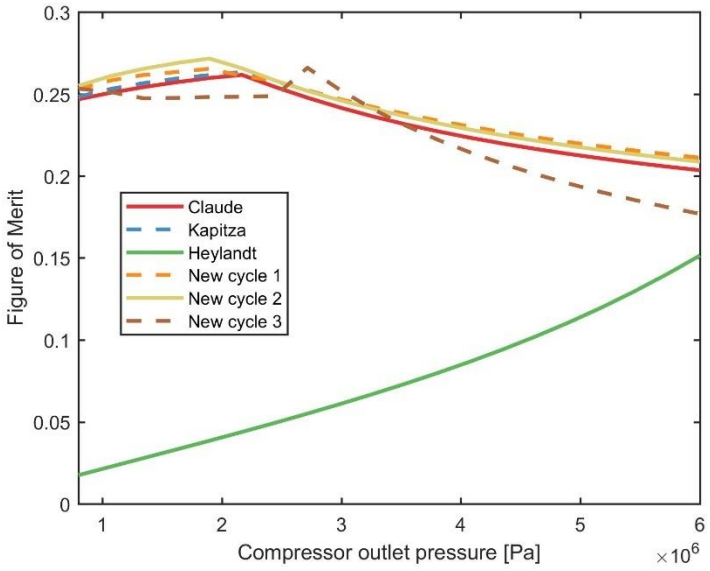


Figure 3.23 - Pressure against FOM (for X optimum values)

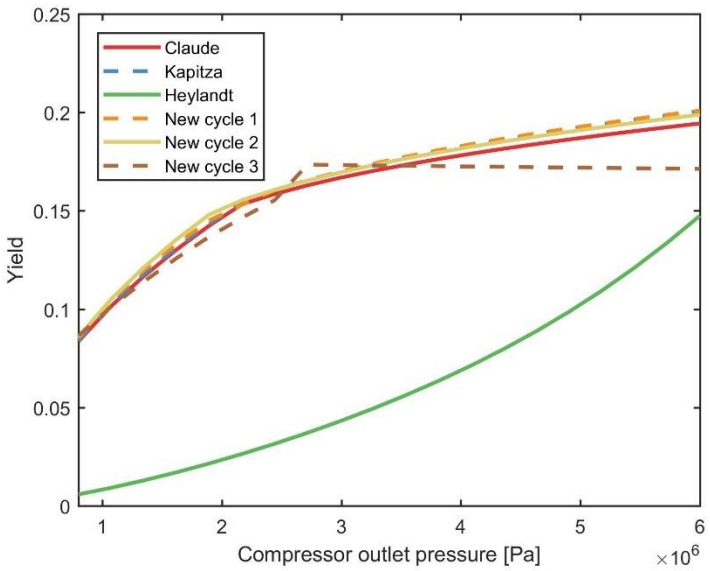


Figure 3.24 - Pressure against yield (for optimum X values)

3.11. Effect of Component Effectiveness/ Efficiency Values

The following results allow the effect of the component efficiencies to be analysed. This can allow both lower limits for the efficiencies to be defined and to determine which component's efficiency has the largest effect on the whole cycle FOM and therefore, which component should receive more attention when designing a system.

3.11.1. Heat Exchanger Effectiveness

When looking at the high-pressure cases shown in Figure 3.25, reduction in the heat exchanger effectiveness down to 0.85, for New Cycle 3, does not result in a large drop in FOM. The others all follow an almost linear trend at effectively the same rate of reduction, with all except the Heylandt cycle being closely grouped.

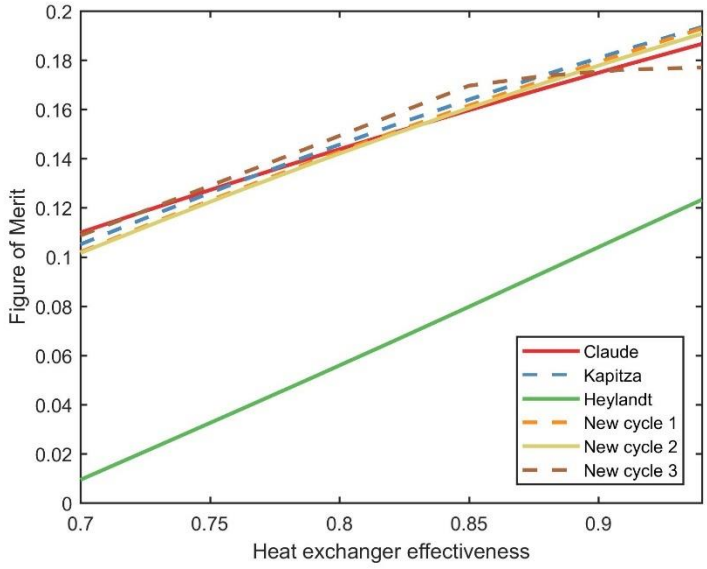


Figure 3.25 – FOM against heat exchanger effectiveness (60bar operating pressure, optimum X values)

In the low-pressure cases, the effectiveness can be seen to have a much greater effect. This can be seen in Figure 3.26, where the trends are still fairly linear, with a much steeper gradient. Because of this gradient, all the cycles have stopped liquefying gas when the effectiveness drops too low. It can therefore be concluded that heat exchanger effectiveness is much more important for low pressure systems.

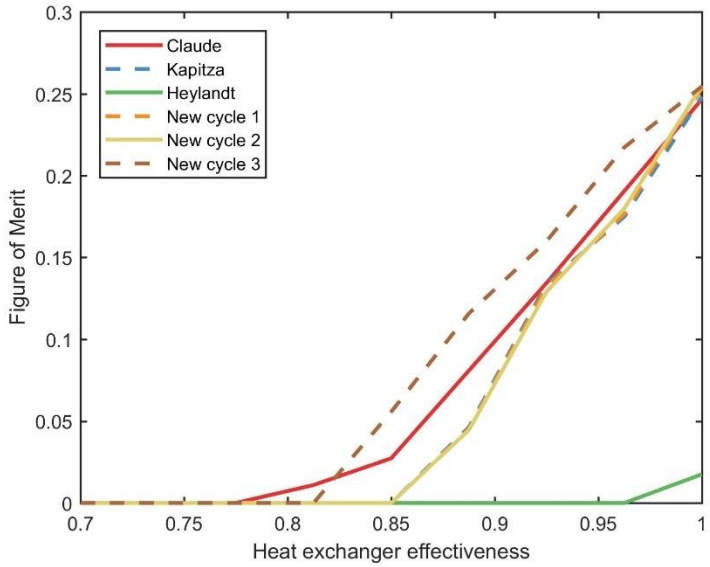


Figure 3.26 - FOM against heat exchanger effectiveness (8bar operating pressure, optimum X values)

3.11.2. Compressor Isentropic Efficiency

The compressor isentropic efficiency affects the FOM very linearly as well, as can be seen in Figures 3.27 and 3.28. This differs from the effect of the heat exchanger effectiveness in that at low pressures the reduction in FOM has very similar gradient to the high-pressure condition.

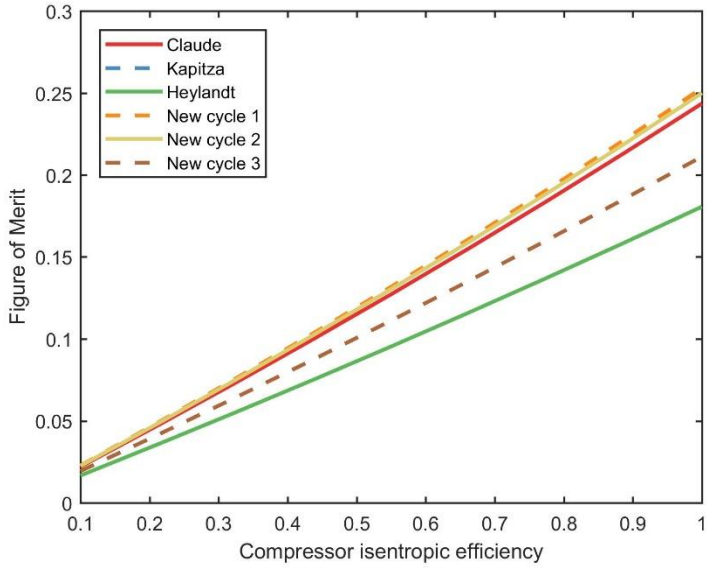


Figure 3.27 - FOM against compressor efficiency (60bar operating pressure, optimum X values)

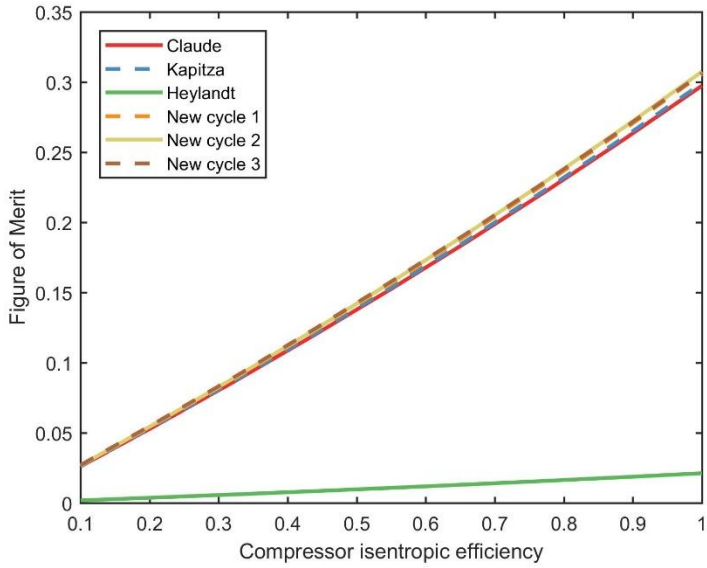


Figure 3.28 - FOM against compressor efficiency (8bar operating pressure, optimum X values)

3.11.3. Expander Isentropic Efficiency

The analysis of the expander's efficiency shows interesting trends at high values, the increase in FOM levels off and becomes much less/negligible with all but the Kapitza and Heylandt cycles, this is shown in Figure 3.29. This is due to the expander reducing the temperature of the gas in cycle so much that

it is close to liquifying before passing through it. Therefore, it liquefies early inside the expander, reducing the work output, as it becomes an incompressible liquid and no longer expands. This effect could be prevented if the flow diversion values were adjusted, this would require the multi-parameter optimisation (Parameters of expander efficiency, X1 and X2), similar to that shown previously. This would be a computationally heavy process.

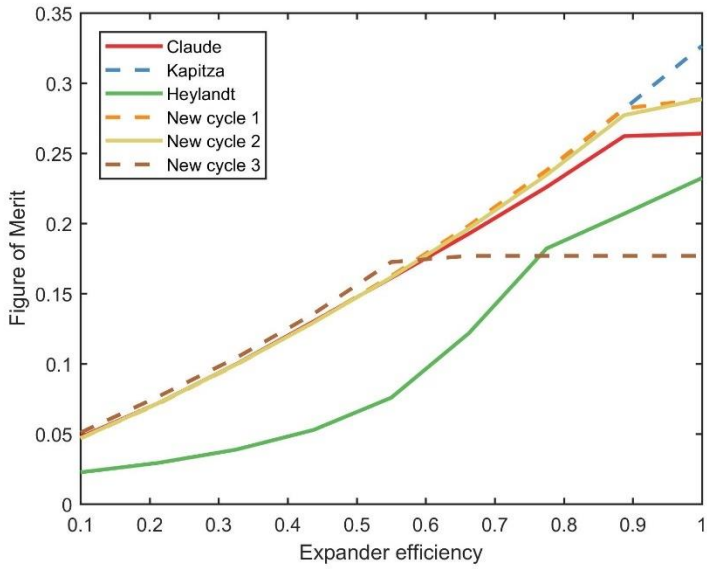


Figure 3.29 - FOM against expander efficiency (60bar operating pressure, optimum X values)

The low-pressure cases in Figure 3.30 show that all cycles benefit with exponential trends. It does not exhibit the same halting in gradient as the high-pressure cases before. This is because the pressure difference does not provide enough cooling to the flow (that will cycle around to cool the flow before the expander) to bring the expander's own inlet to liquid/near liquid conditions.

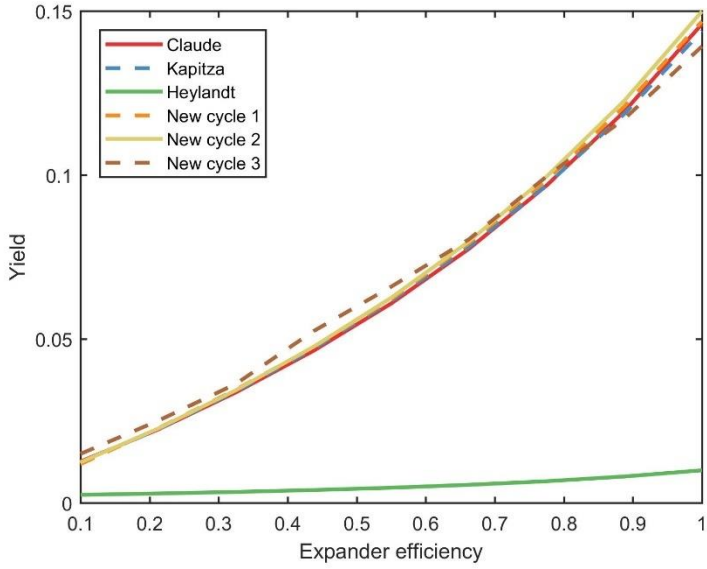


Figure 3.30 - FOM against expander efficiency (8bar operating pressure, optimum X values)

3.11.4. Discussion of Component Efficiencies

The impact of the compressor's efficiency and the heat exchangers' effectiveness values on the yield and FOM of the cycles are generally linear in nature. The impact of the expander's efficiency value is generally exponential in nature. Therefore, a decreased expander efficiency value will result in a much greater loss in cycle yield and FOM in comparison to the same drop in compressor efficiency or heat exchanger effectiveness. Consequently, the expander is shown as most important component in achieving the best yield and FOM for these cycles and will be the focus for the remainder of this thesis.

3.12. Chapter Summary

- Three well known gas liquefaction cycles and three modified versions of these were described.

- MATLAB models were created to simulate the steady state operation of all six cycles.
- The MATLAB models used an iterative approach to converge on a solution.
- The models were used for a parametric study to analyse how the cycles performed at different pressures, flow diversion fractions (X), and component efficiencies.
- The optimum flow diversion fractions (X) were found to be different for high-pressure and low-pressure operating conditions for each cycle.
- For most cycles (except Heylandt) there was a peak FOM at an operating pressure of about 20bar, however the yield continued increasing with pressure for all cycles.
- For all cycles, the heat exchanger and compressor efficiencies affected the FOM close to linearly, whereas the expander efficiency affected the FOM in an exponential way. Therefore, the expander efficiency is the most crucial for good cycle performance.
- New cycle 3 gave a slight improvement (FOM of 0.30) over the other cycles, the Kapitza cycle (FOM of 0.28) is only slightly lower but is a much simpler design. Therefore, it may be recommended as the better option.

CHAPTER 4

WANKEL EXPANDER CFD SIMULATIONS: DESIGN AND SETUP

4.1. Chapter Introduction

The literature review chapter showed that many small-scale thermodynamic systems could benefit from a low-cost, high efficiency expansion device. The Wankel expander was shown to be simple and cheap to manufacture, giving it low cost, and the potential to have high efficiency and power. However, to achieve the high efficiency performance, external valves had to be used, increasing the size, complexity, auxiliary power losses and ultimately, the cost. Another problem the Wankel expander is often cited to have is friction and lubrication problems from the seals. The working principle of a standard Wankel expander, without external valves, is shown in Figure 4.1 and the geometry sections are labelled in Figure 4.2.

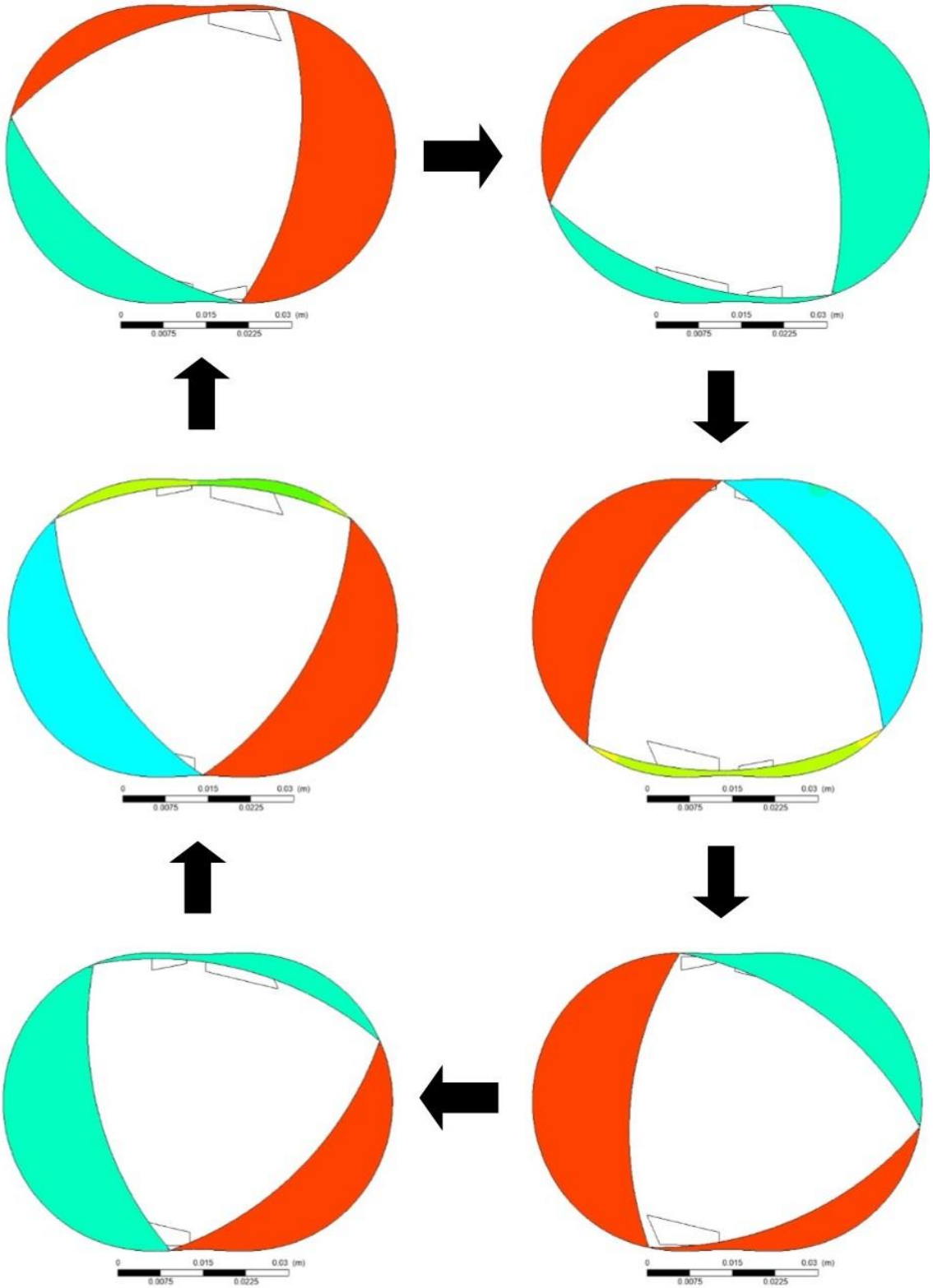


Figure 4.1 -Cycle diagram of the standard Wankel expander (CFD pressure contours)

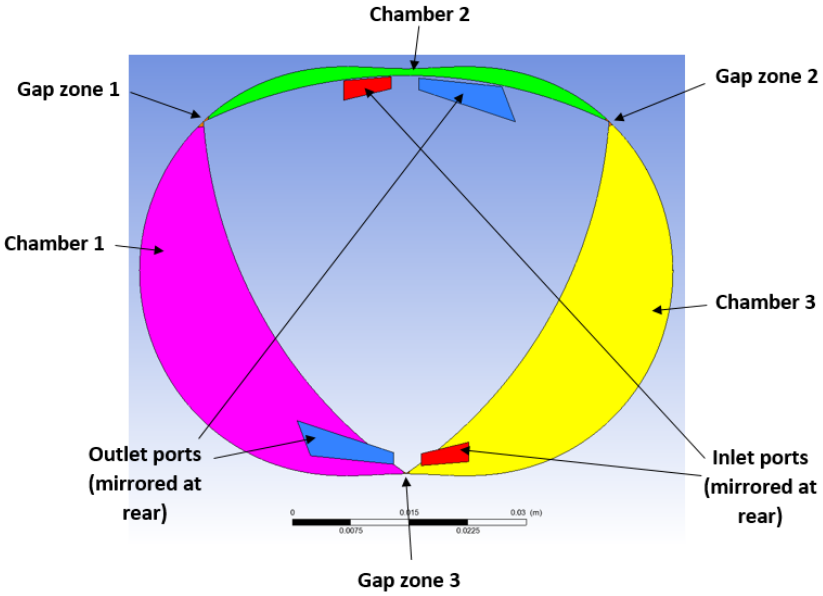
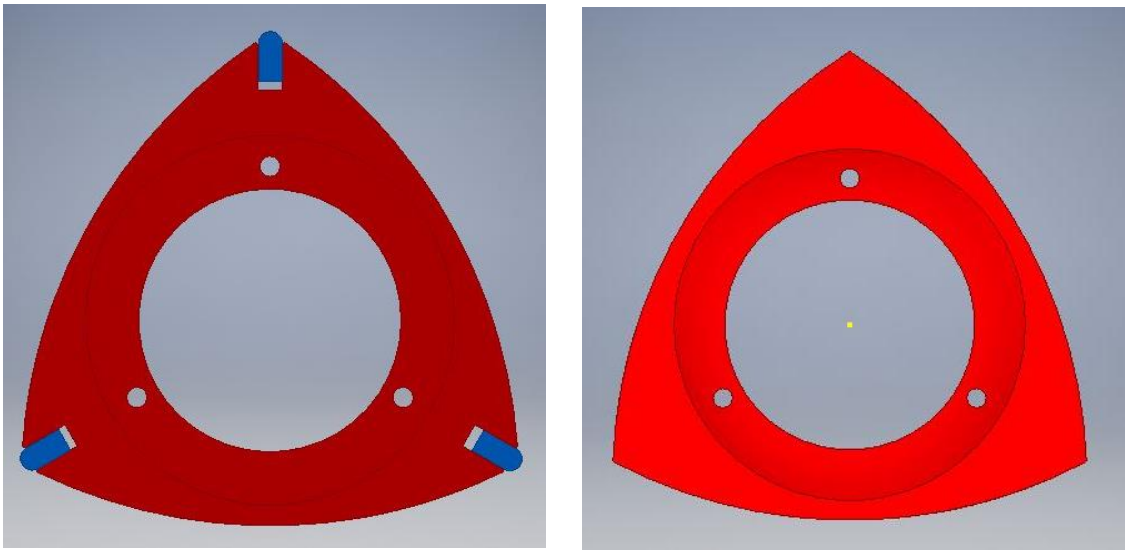


Figure 4.2 – Standard Wankel expander's geometry and port configurations

4.2. Wankel Expander With and Without Apex Seals

The seals in the Wankel expander have been shown to contribute a larger proportion of its performance losses. They also cause problems with extra wear, lubrication, complicating the assembly process, and increasing the maintenance of the expander. One further application specific problem is that they produce heat through friction, which would be counterproductive for an expander being utilised for its cooling effect, such as in a liquefaction or refrigeration system. Therefore, it could be advantageous if they were removed. Figure 4.3 shows a comparison of a Wankel expander's rotor with and without apex seals. The case with apex seals utilises leaf springs to press the seals against the housing wall.



(a) (b)

Figure 4.3 – CAD of a Wankel expander rotor (a) with apex seals and (b) without apex seals

The problem with removing the seals is the increased leakage at the interface of the moving parts which will cause loss in the expander's performance. Therefore, an analysis is carried out to predict the performance of the Wankel expander with and without apex seals to predict the performance loss due to leakage. Furthermore, analysis is required on different apex clearances in the seal-less case, to ascertain which apex clearances have acceptable performance and what size clearance machining and bearing tolerances will allow.

A Wankel design without apex seals has been suggested before as a pump in literature [102]. However, as the liquids in a pump application have a much greater viscosity, the size of the apex clearances are less important. Oil flooding the gas working fluid could provide a similar advantage for a gas compressor or expander. However, an added oil system would increase the device size, complexity, and axillary power losses. Therefore, there is a

possible advantage to the seal-less design without oil flooding, even though the isentropic efficiency of the device will be affected by the increased leakage. Furthermore, comparing cases with and without apex seals directly is fair, as the cases with apex seals require oil for lubrication anyway.

4.3. Static Shaft Wankel Expander

The second issue to be analysed is the inlet timing control. In order to operate with maximum efficiency, the Wankel expander requires a valve system in order to cut off the inlet flow and prevent under-expansion. Felix Wankel's first Wankel engine design, named the DKM Wankel, operated differently to the most common version these days. This design had the disadvantage of a rotating housing, making it difficult to design and assemble with engine sparkplugs. However, this is not a problem in a gas expander as the lack of combustion means sparkplugs are not required.

This design offers multiple other advantages. As the Housing and rotor both rotate around their own centres of mass, they are both dynamically balanced and the design therefore does not need additional balancing masses. This reduces complexity, size, and weight. The design also has the inlet flow routed through the shaft in such a way that it can be controlled much better than the standard Wankel expander, without the need for an extra valve system. This also means that the exterior valve system that would have previously been required, is not, and the complexity, size, and auxiliary losses are reduced in that way.

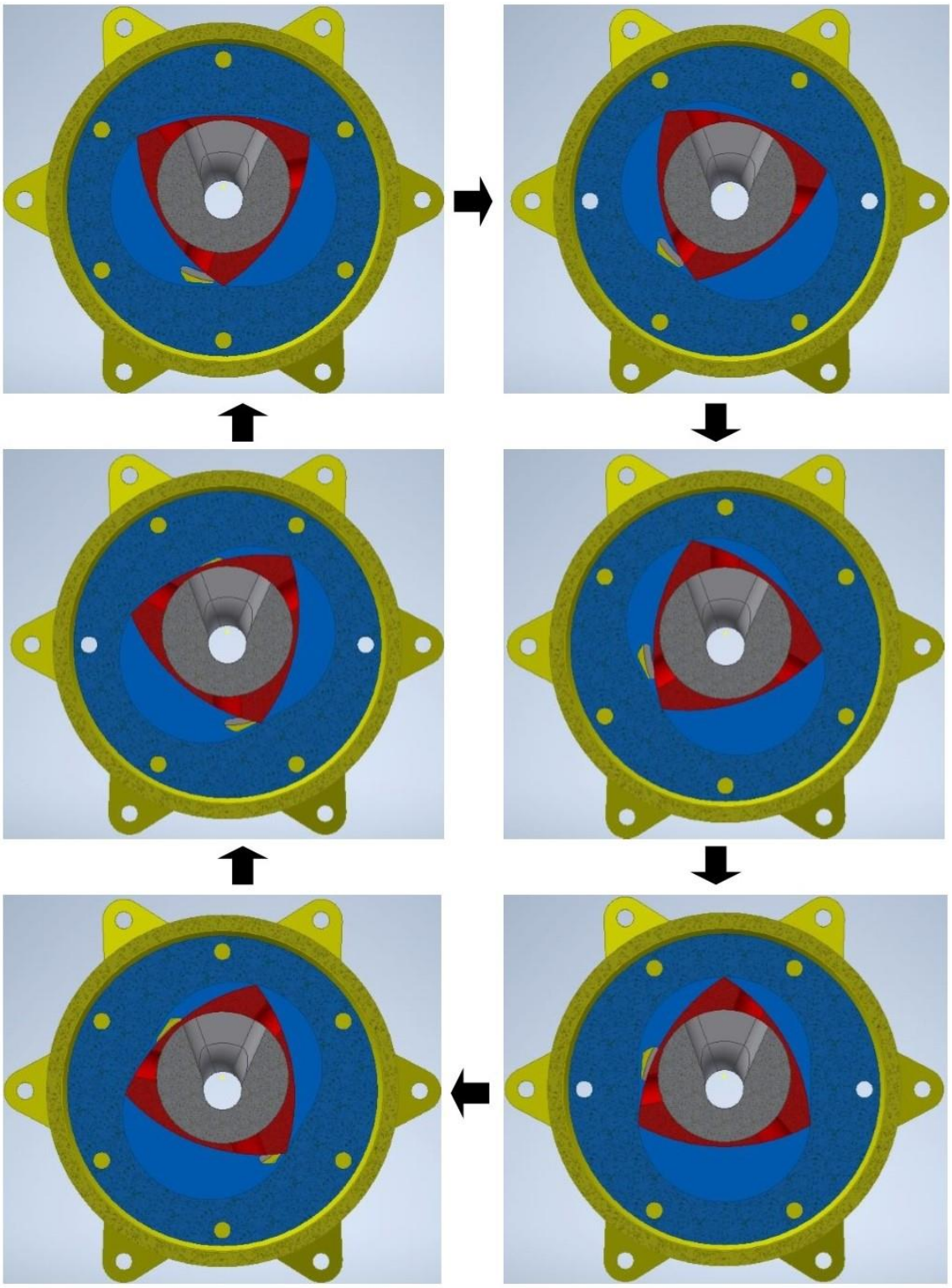


Figure 4.4 – Cycle diagram of the static shaft Wankel expander (CAD prototype design)

Due to these advantages, a Wankel gas expander variant of this concept was designed, the working principle can be seen in Figure 4.4 and the geometry

can be seen in Figure 4.5 in the form of CFD pressure contours, it can be compared to the standard Wankel expander's side ports in Figure 4.2. The new design was named the static shaft Wankel expander. The first key change with the static shaft Wankel expander is the centres of rotation for the rotor and housing are fixed, which keep the device inherently balanced during operation.

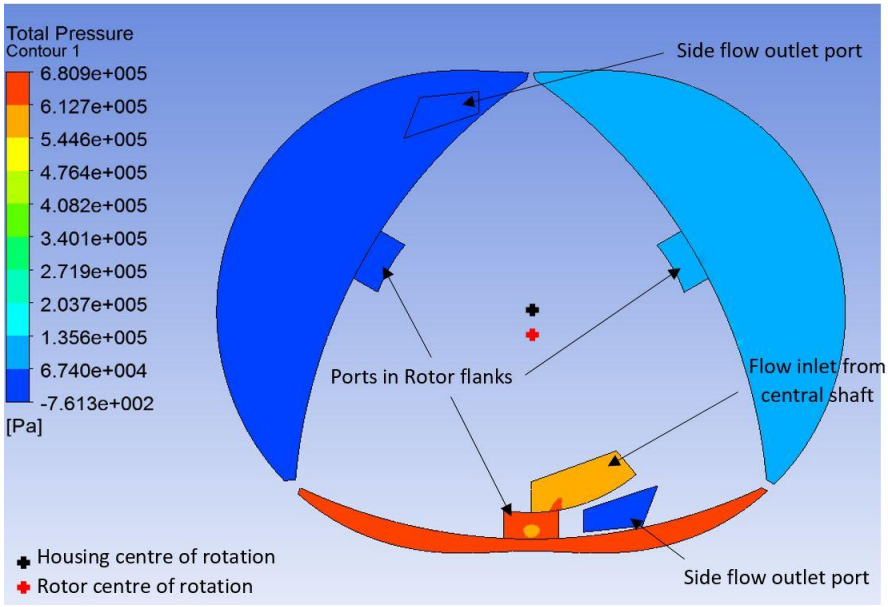


Figure 4.5 – Static shaft expander's porting configuration and rotation axes

The second change is that the inlet ports are moved to the rotating rotor flanks, which permit inlet to each chamber when they align with the stationary central shaft inlet. To better understand how the inlet flow enters the new design, a cross section of a CAD model of the expander is shown in Figure 4.6, where the red arrows show the path of the inlet gas flow. The gas travels through the grey inlet shaft and then through a hole in the red rotor into the chamber. To better try and display the dynamic nature of the inlet port opening and closing, 3D CFD streamlines are shown in Figure 4.7 at two

different times in a revolution. Note that the inlet shaft has been reduced to the small section that interfaces with the rotor inlet ports to reduce simulation time. It can be seen how the port in the rotor flank moves around the stationary inlet shaft port, therefore the length of this inlet shaft port arc can be used to change the inlet opening time for the device.

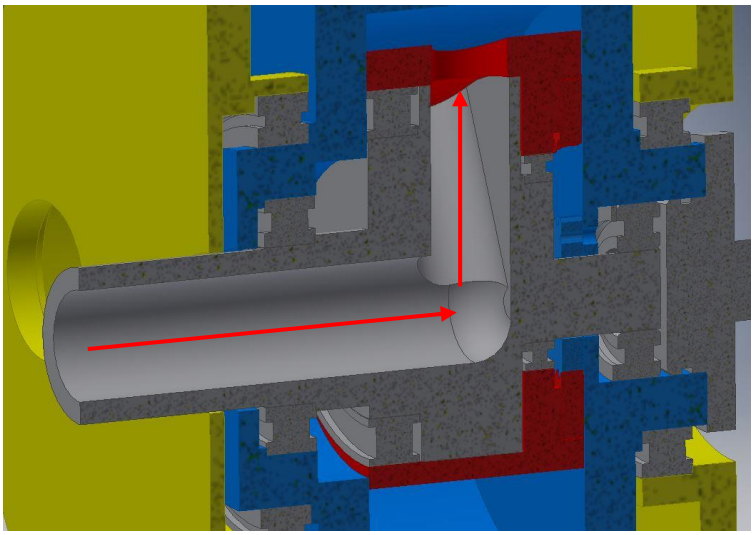


Figure 4.6 – Inlet flow through shaft

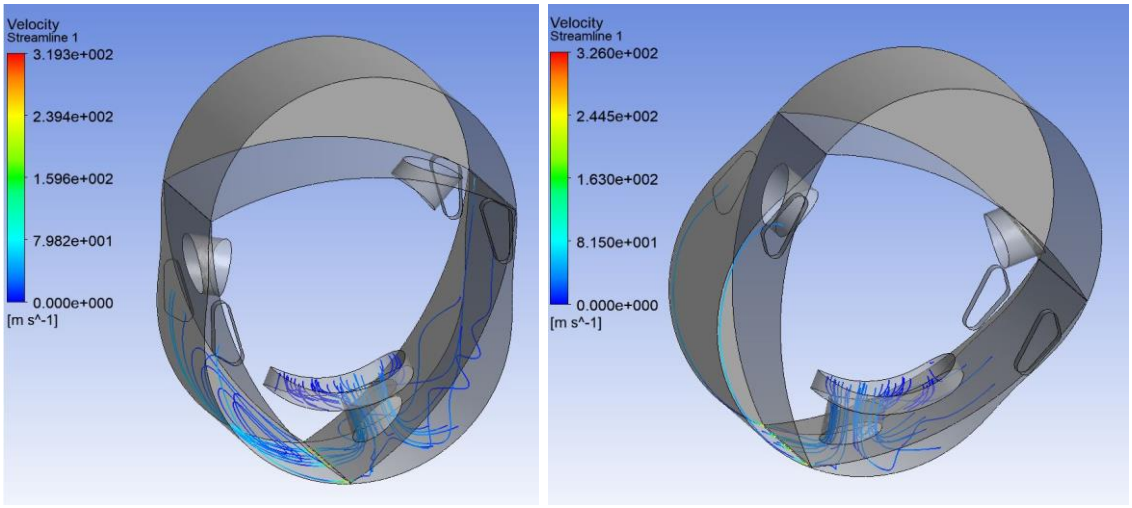


Figure 4.7 – Flow streamlines from the inlet port at two times

Two disadvantages exist for this setup, the first being that a separate outer housing is required to capture the outlet gases as the outlet ports are rotating.

The second is that two extra bearings are required when compared to the standard design, which will increase mechanical losses. Analysis is therefore required to ascertain whether the advantages outweigh the disadvantages.

4.4. Introduction to CFD

CFD is a method of computationally analysing how fluid acts in a 2D or 3D environment, defined by a digital geometry representing the real world. The process of setting up CFD simulation is presented by a flowchart in Figure 4.8 and in more detail in the following text.

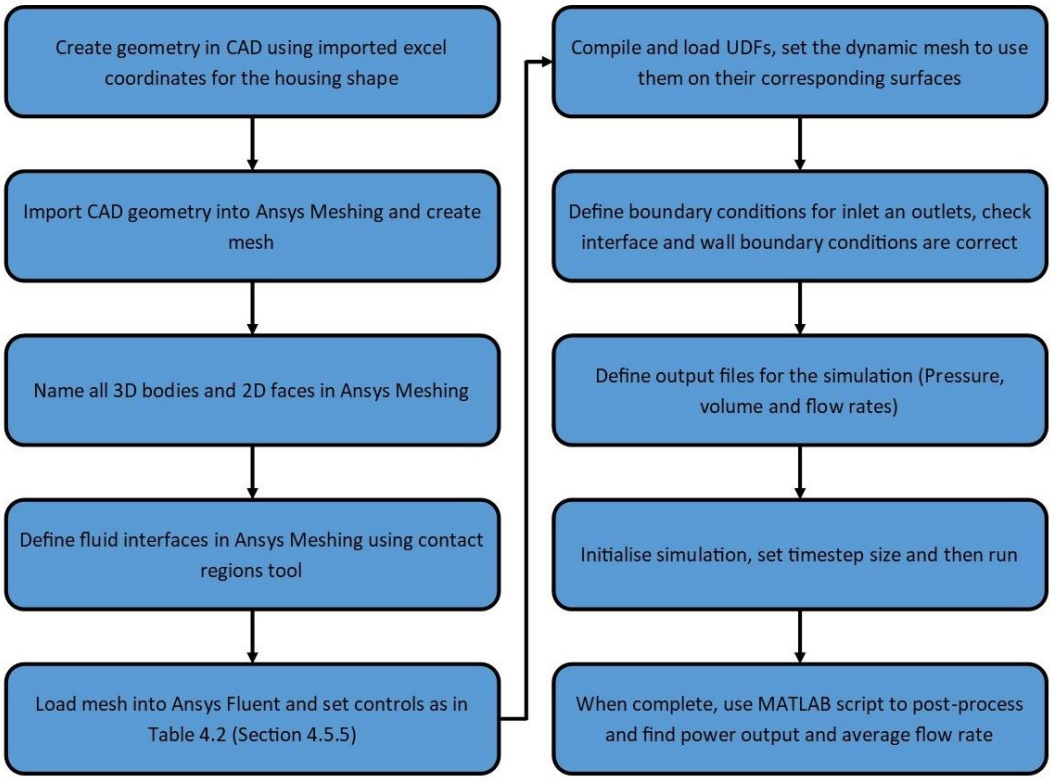


Figure 4.8 – Flowchart of the setup of a Wankel expander CFD simulation

A CFD simulation starts with a created mesh of cells, which approximate the geometry of the fluid to be modelled. A cell within the mesh can be of many different shapes and is defined by its surrounding cell walls, wall edges and

cell nodes where the edges meet. The smaller the cell size (and the more cells in a mesh) creates a closer approximation to the required geometry, however as each cell requires its own equation calculations, the processing time of the simulation increases with more cells.

Each cell in a mesh contains information of the fluid that is encapsulated within its volume, including but not limited to the volume size, the pressure, the temperature, and the velocity. Before the model is solved, boundary conditions and initial conditions are set for all the cells. Boundary conditions most commonly occur at the outer limits of the mesh, in the form of constant pressure, constant velocity and/or constant temperature, plus multiple other definitions. Initial conditions are applied to all cells and can either be one condition for the entire mesh, have a linear gradient of conditions between boundary conditions, or be completely user defined for different areas.

Once the mesh is prepared and the model's initial and boundary conditions are set, it is ready to solve. In this process, a series of mass, momentum and energy conservation equations are solved for each cell (see Equations 4.1, 4.2 and 4.3, from the Fluent help manual [109]). Other equations may also be solved if more complex fluid dynamic or thermodynamic phenomena are being modelled, in the case of this study the only other equations used were for turbulence modelling (see Equations 4.4 and 4.5), from the Fluent help manual [109]). The equations take the form of partial differential equations and therefore each cell solution relies on the parameters and gradients of the cells surrounding it. Because the solutions for cells surrounding a particular

cell will also depend on that cell in turn, there are many feedback loops in the solution, so that multiple solving iterations are required to converge on a steady solution. The model stops solving when the solution is considered to be converged, which is usually determined through monitoring residuals of the solution parameters. However, the model can also finish if the user set maximum iterations is reached or the solution parameters are found to be diverging.

Mass continuity equation:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{v}) = 0 \tag{4.1}$$

Momentum conservation equation:

$$\frac{\partial}{\partial t}(\rho \vec{v}) + \nabla \cdot (\rho \vec{v} \vec{v}) = -\nabla p + \nabla \cdot (\bar{\bar{\tau}}) \tag{4.2}$$

Energy conservation equation:

$$\frac{\partial}{\partial t}\left(\rho h - p + \rho \frac{v^2}{2}\right) + \vec{v}\left(\rho h + \rho \frac{v^2}{2} + p\right) = \nabla \cdot \left(k_{eff} \nabla T - \sum_j h_j \overrightarrow{J_j} + \left(\bar{\bar{\tau}}_{eff} \cdot \vec{v}\right)\right) \tag{4.3}$$

Where ρ is density, $\vec{v}$ is the velocity vector, $\bar{\bar{\tau}}$ is the stress tensor, h is sensible enthalpy, p is static pressure, v is total velocity, k_{eff} is the effective conductivity, J_j is the diffusion flux of species j, $k_{eff} \nabla T$ is the energy transfer due to conduction, $\sum_j h_j \overrightarrow{J_j}$ is the energy transfer due to species diffusion and $\left(\bar{\bar{\tau}}_{eff} \cdot \vec{v}\right)$ is the energy transfer due to viscous dissipation.

k-ε turbulence equations:

$$\frac{\partial}{\partial t}(\rho k) + \frac{\partial}{\partial x_i}(\rho k u_i) = \frac{\partial}{\partial x_j}\left[\left(\mu + 0.09\rho\frac{k^2}{\varepsilon}\right)\frac{\partial k}{\partial x_j}\right] + G_k + G_b - \rho\varepsilon - Y_M \quad (4.4)$$

$$\frac{\partial}{\partial t}(\rho\varepsilon) + \frac{\partial}{\partial x_i}(\rho\varepsilon u_i) = \frac{\partial}{\partial x_j}\left[\left(\mu + \frac{0.09\rho\frac{k^2}{\varepsilon}}{1.3}\right)\frac{\partial \varepsilon}{\partial x_j}\right] + 1.44\frac{\varepsilon}{k}(G_k + C_{3\varepsilon}G_b) - 1.92\rho\frac{\varepsilon^2}{k} \quad (4.5)$$

Where μ is dynamic viscosity, G_k is the generation of turbulence kinetic energy due to mean velocity gradients, G_b is the generation of turbulence kinetic energy due to buoyancy, Y_M is the contribution of the fluctuating dilatation in compressible turbulence to the overall dissipation rate, $C_{3\varepsilon}$ is a constant.

When a solution is converged, the conditions at each of the mesh cells are saved. These can then be processed by the user to analyse many different parameters, such as pressure, temperature, or velocity contours, streamlines of flow or velocity vectors. These can be viewed in 2D or 3D and can also be seen as animations. Furthermore, numerical outputs can be calculated to find the values of various conditions at any location in the expander [109].

4.5. CFD setup

CFD was decided upon as an analysis tool for the Wankel expander. This is because the Wankel expander has both complex geometry and motion. This causes the expansion chamber walls to be in constant motion in relation to the enclosed gas and additionally, the shape and surface area are constantly changing in a non-linear way as shown in Figure 4.9. Furthermore, the shape and size of the inlet and outlet ports control the timings and can therefore take the form of relatively complex shapes. The flow through complex shaped

ports would be very hard to calculate accurately without CFD or experimental work. When CFD is utilised, all of these factors can be calculated simultaneously.

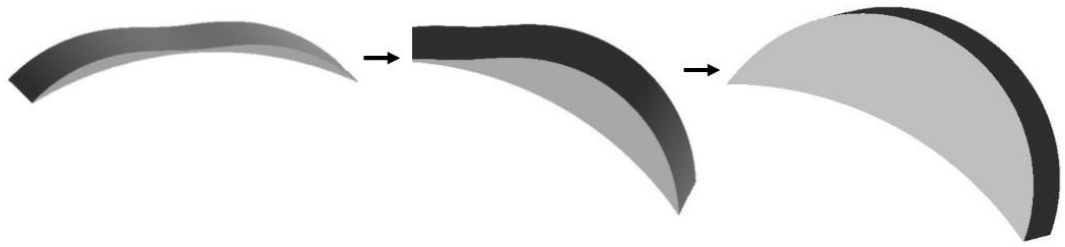


Figure 4.9 – Different shapes of an expansion chamber throughout a cycle

4.5.1. CFD Assumptions

As with all CFD, certain assumptions had to be made to reduce computational time. These assumptions will reduce the accuracy of the model and therefore the effect of the assumption should be understood. Table 4.1 shows the assumptions made in the CFD and the reasons they were made.

Table 4.1 – Assumptions made in the CFD analysis

Assumption	**Reason**
Ideal gas equations used	For the conditions used in the simulations, this provides perfectly accurate results.
Constant pressure inlet and outlet boundary conditions	This would be very close to reality as the expander was installed in a steady state system.
No slip wall boundary conditions	We are not concerned with near wall flow conditions and more on the performance of the device as a whole.
Expander speed	The rotational speed was modelled as a constant speed, although this would not be the case in reality, the actual speed fluctuations would be insignificant at the speeds analysed.
Friction	Bearing friction was not accounted for in the calculations as at the time, it was unknown how many and what type of bearings would be used.
Heat transfer	Heat transfer through walls could not be defined as the depth of the walls, material and outside conditions could not be guessed at this stage, therefore zero heat transfer (adiabatic) wall conditions were used.
Side face leakage	Including side face leakage in the CFD simulation would increase simulation time excessively, therefore it was ignored. It can possibly be accounted for in a semi-empirical model once experimental data is acquired.

4.5.2. CAD Design and Meshing for CFD

The first step in setting up the CFD model is creating the geometry. The geometry needs to be created as 3D solids in the form that the fluid will take. The software used to create this is Autodesk Inventor after which each constituent part is saved as an IGES file. The three expansion chambers are created using imported series of coordinates for the housing faces and three arcs of radius 65mm for each rotor face. The housing coordinates are created in a Microsoft Excel spreadsheet using Equations 4.6 and 4.7.

$$x_h = e \cdot \cos(3\theta) + r \cdot \cos(\theta) \tag{4.6}$$

$$y_h = e \cdot sin(3\theta) + r \cdot sin(\theta) \tag{4.7}$$

Where x_h and y_h are the x and y cartesian coordinates of the housing, e and r are the eccentricity and rotor radius respectively and θ is the generating angle.

The inlet and outlet ports are created as further separate files in Autodesk Inventor. All the 3D bodies are separate files to ensure they are separate bodies when imported into Ansys, where they are required to move with different motions.

Once the geometries of the different parts are created, they are imported into the Ansys meshing software, accessed via Ansys Workbench. The software offers various controls to adjust the mesh to be optimal for the simulation. The first and most important control is setting a tetrahedron dominant mesh, in which the mesh cells are predominantly made from tetrahedron shapes. This form of mesh is desirable for two main reasons. The first is that some dynamic mesh motion tools in Ansys Fluent require this form of mesh to operate. The second is that tetrahedron dominant mesh provides a more accurate solution when the direction of flow is unknown [109] , as aligning the mesh with the flow is no longer a requirement.

Another important pair of mesh control methods are face meshing and mesh sizing. These are used together to form a uniform square grid at each of the apex interfaces. The face meshing control creates square meshing if the selected face is four sided, as shown in Figure 4.10. The mesh sizing allows the user to specify the number of cell divisions along the edges of the face.

This allows the number of nodes across the clearance to be set precisely and the mesh density to be much higher than that of the majority of the expansion chamber. This is necessary because the small apex clearances require the fine mesh density, but the rest of the Wankel expander does not. If the entire mesh density were as fine as the apex clearances, the computational time would be unfeasibly large.

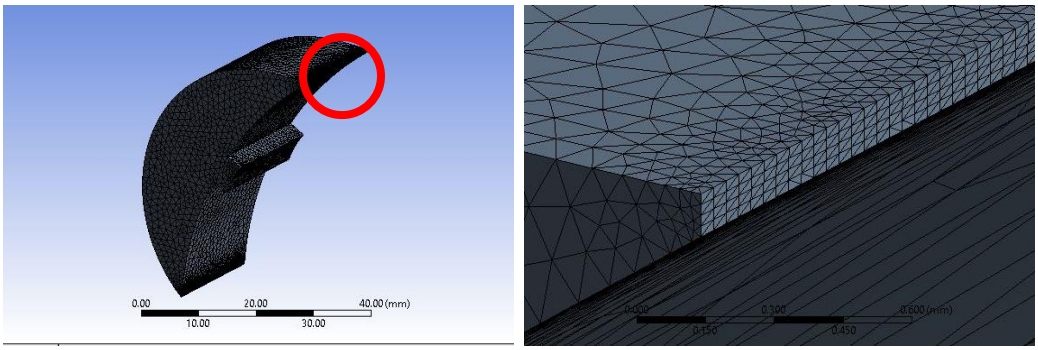


Figure 4.10 -Ordered face meshing at the apex clearance interface between two chambers

The meshing needs to be done in such a way that the size of the boundary mesh faces can control the internal mesh when the mesh movement is produced in Fluent, but also so there is a sufficient mesh density to accurately model the flow. As there are two very different areas of the mesh (the large expansion chamber and the small apex clearances), two mesh sensitivity studies were done to ensure the mesh density in each location was acceptable. Figure 4.11 shows the refined mesh density at a rotor apex. The first study focused on the mesh density of the expansion chamber areas of the expander. The results from the first mesh sensitivity study are shown in Figure 4.12, where it can be seen that after below 1mm maximum cell size the solutions

are stable (less than 0.4% variation in efficiency and power). A maximum cell size of 0.6mm was decided on for the simulations.

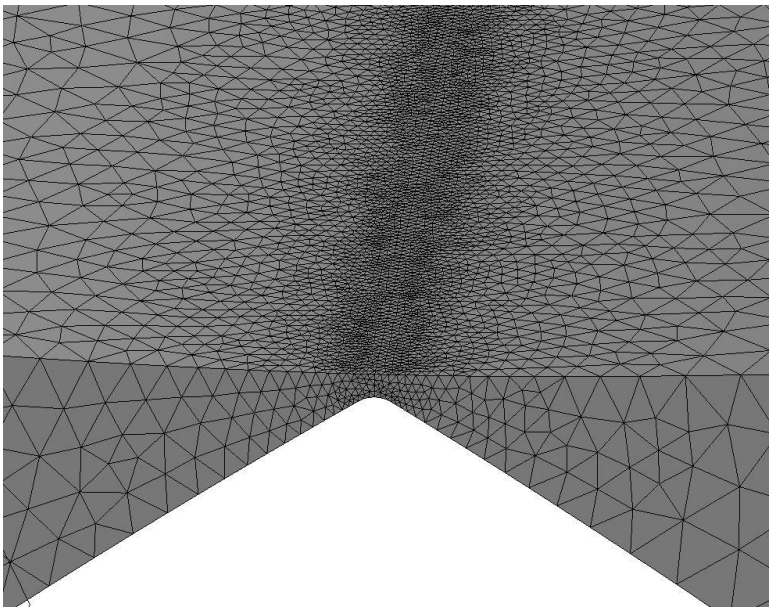


Figure 4.11 – Mesh density variation at a rotor apex clearance

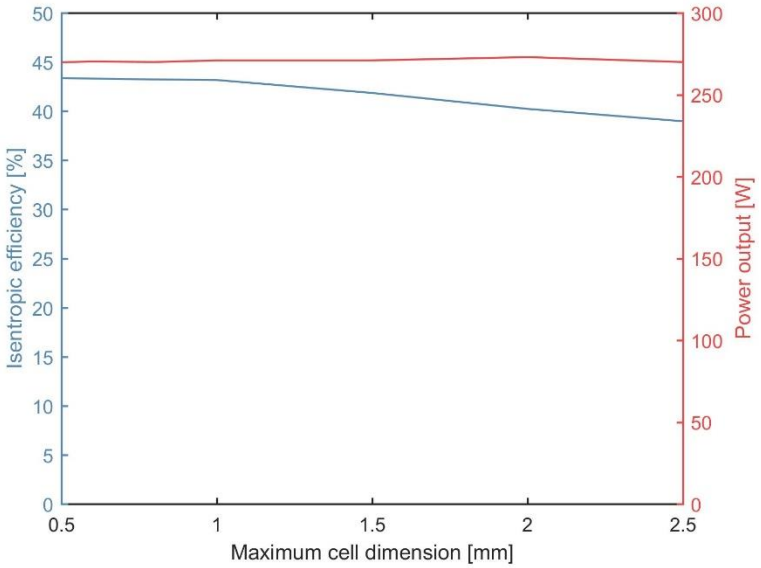


Figure 4.12 – Mesh sensitivity study of chamber cell sizes (4barg inlet pressure, 4800rpm,0.1mm apex clearance)

The second mesh sensitivity study shown in Figure 4.13 helps decide the mesh density at the rotor apex locations. The number of nodes across the apex

clearances is shown to be not overly important, as changing them from the minimum possible to a large amount, produced a small change. This change is small enough (0.261% variation for power output and 0.558% for efficiency) that other sources of error produce a much larger variation. The number of nodes across each apex clearance was set to five, as this is a low enough number that the computational effort is not overly increased. A number less than this was not used as, even though they produced sufficiently accurate results, they were more susceptible to simulation failures at certain operating conditions.

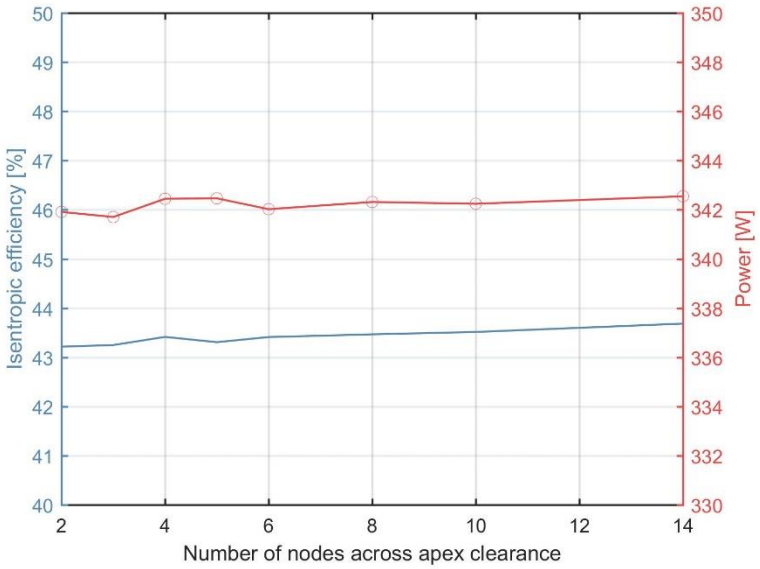


Figure 4.13- Mesh sensitivity study of the number of nodes across the apex clearances

Before the mesh is complete, the separate 3D bodies and faces must be named and the interfaces between different sections must be defined. Naming the different bodies and faces of the mesh allows for easier definition of the interfaces, boundary conditions and dynamic mesh motion. Specifying the interfaces tells the Ansys Fluent software which mesh faces fluid is permitted

to flow between when they are in contact. This is crucial for the valve like nature of the inlet and outlet ports.

4.5.3. UDFs For Mesh Motion

To create the motion of the Wankel expander in the transient simulation, User Defined Functions (UDFs), which are a provision of ANSYS Fluent, are utilised. UDFs provide the functionality to control many aspects of the simulation model via code written in C language. The UDFs are required as the complex motion of the Wankel expander cannot be defined by the built-in mesh motion functions in Ansys Fluent, which generally only allow for simple rotational or translational motion of rigid mesh bodies.

The UDF functionality provides various 'macros' that can be used to link the code between the UDF and Ansys Fluent. The different macros can allow data to be retrieved from the simulation, controls to be changed, data to be altered plus many other things. The macro used for the following models is called 'Define Grid Motion', which allows the user to define how the mesh moves in a transient simulation. The macro gives the user the time and the time step as an output from Ansys Fluent. The user can then use a further function to cycle through all the cell nodes in a given domain and manipulate the location of each node individually.

Once the UDF is written in the C code, it must be compiled in the Ansys Fluent software, as it makes use of macros which are not part of the base C language. After compilation, the UDF appears in a drop-down menu in the

dynamic mesh setup section of Fluent. In this section you can apply any dynamic mesh UDF motion to any face or body of the mesh. The C code of the UDFs written for the following simulations include global constants which must be set before compilation. These constants include the dimensions of the Wankel expander and the speed of rotation.

In this research, two different Wankel expander concepts that each require different motion types have been studied. The first type, named the 'standard Wankel expander', is the common Wankel design in which a Rotary piston rotates and simultaneously follows a circular translation inside a stationary housing. The second type, named 'Static Shaft', involves the rotary piston just rotating about its own centre of mass and the surrounding housing also rotating, but around a different axis. These were both described in detail at the beginning of this chapter.

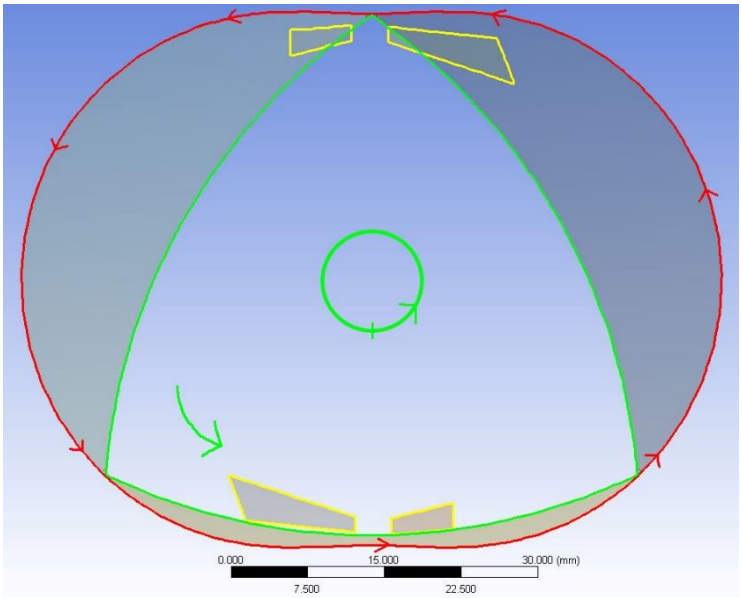


Figure 4.14 – Movement of the nodes in the standard Wankel configuration (Yellow are stationary parts, green are rotor walls and their movement and red is the housing wall's movement)

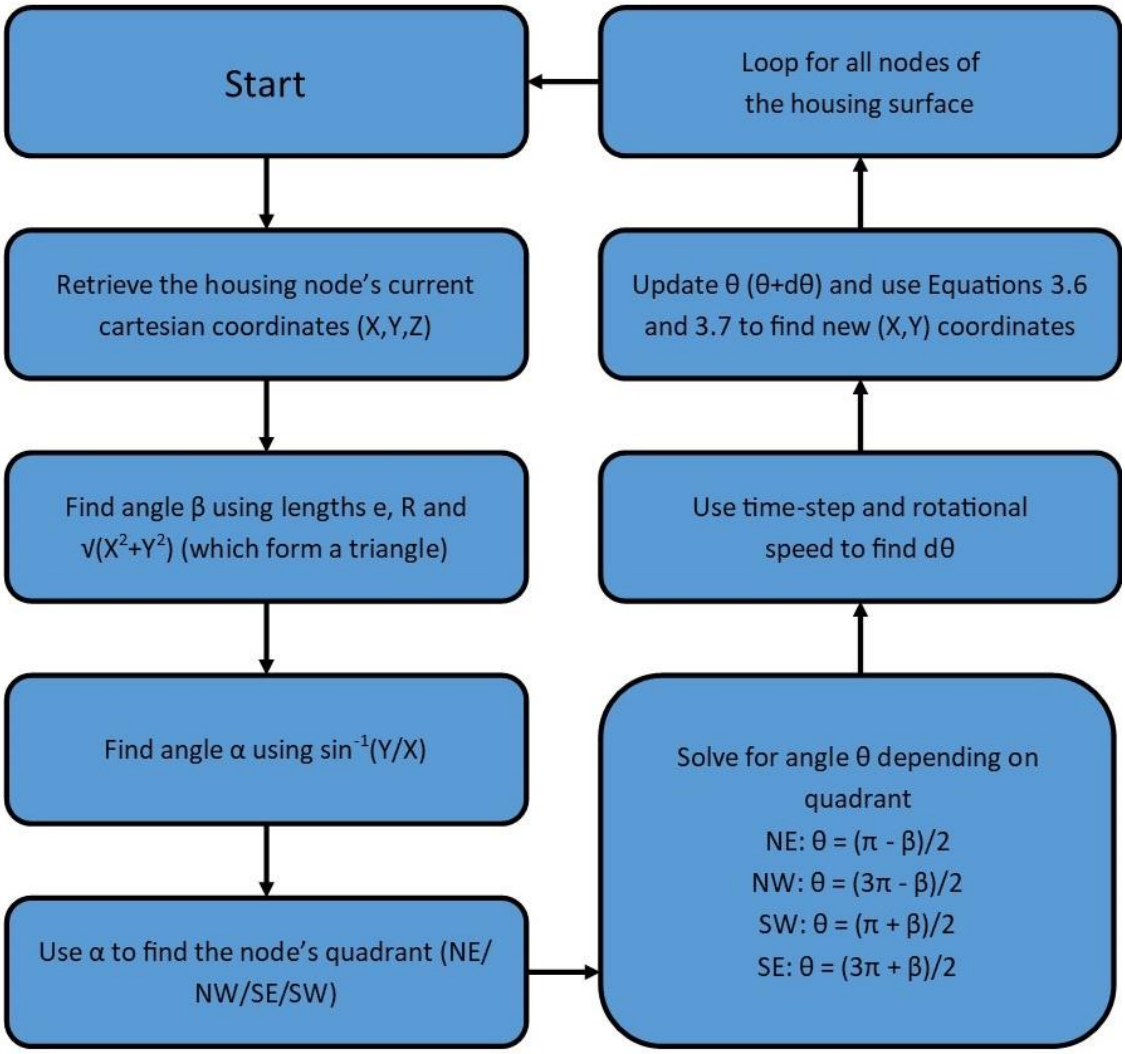


Figure 4.15 – Housing UDF flow chart

Two UDFs are required for the standard type, one which moves the mesh nodes of the rotor faces in the eccentric rotational movement, shown in Figure 4.14 in green. This is achieved by first rotating the mesh nodes the correct amount around the rotation axis (centre of rotor) and then translating the rotation axis the correct amount around the defined eccentric circle. The second UDF translates the nodes of the housing face around the housing face in a 'belt' like motion, whilst maintaining the shape, shown in red in Figure 4.14. This is necessary as the point where the rotor apex meets the housing

face is constantly moving during the simulation, therefore the housing mesh nodes do not also move at the same speed, the cells at the interface would become too skewed and cause a failure. The housing UDF is described by the flow chart and diagram in Figures 4.15 and 4.16.

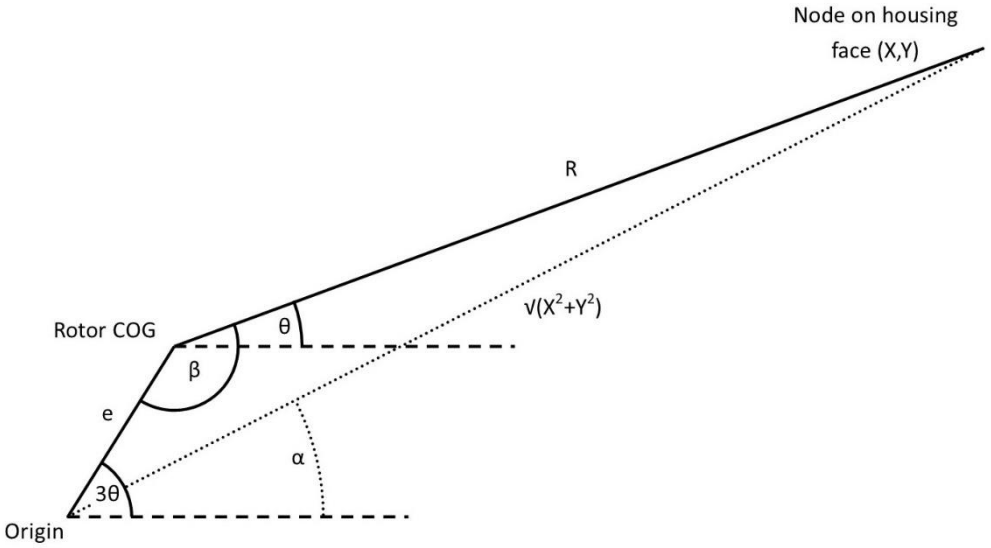


Figure 4.16 – Angles and lengths used for the Housing UDFs

As the nodes of the housing are moved in a way that the housing of an actual expander would, steps should be taken to ensure this does not affect the accuracy of the model. This can be achieved through setting the velocity of all fluid at the housing walls to zero in the absolute reference frame. However, it was found that this generally had no discernible effect on the results, this is due to the very low viscosity of air, meaning the boundary layer thickness is tiny in comparison to the surrounding volume of fluid.

Three UDFs are required for the static shaft Wankel expander design, and they are very similar to the UDFs mentioned previously. The difference with the rotor UDF is that the axis no longer translates on a circular path,

therefore only rotation occurs as shown in green in Figure 4.17. The second UDF, like the standard expander before, moves the nodes on a 'belt' like path around the periphery of the housing, but additionally it rotates the nodes about the housing's centre axis, shown as red lines in Figure 4.17. The third UDF simply rotates the outlet ports around the origin to keep them in the same relative position on the side of the housing as it rotates, shown in blue in Figure 4.17.

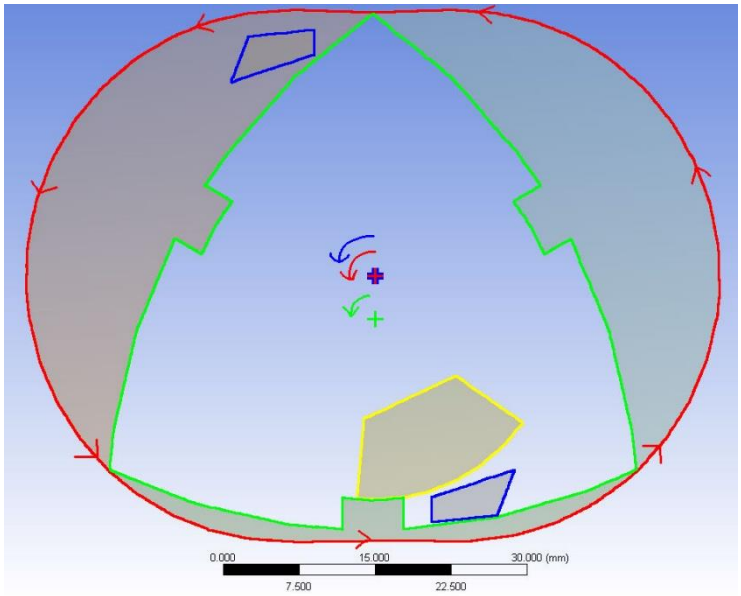


Figure 4.17 – Movement of the nodes in the Static Shaft Wankel configuration (Yellow is the stationary shaft inlet port)

4.5.4. Fluent Dynamic Mesh Setup

Aside from the mesh motion controlled by the UDFs, there are some mesh controls that need to be utilised within the Fluent software itself in order to maintain the correct mesh shape, size, and quality for accurate results. The first of these controls are those that are used on the Wankel expander's side housing walls. As these walls are always parallel to the x-y plane, they must

be restrained in the z axis. Because they are flat faces, Fluent provides a 'Plane' deformation option that can be used. This option allows the internals of a mesh face to deform in any way, provided the nodes remain on a specified 2D plane. It maintains the quality of mesh face using two methods called smoothing and remeshing. These two methods are also used inside all deforming volumes to maintain a good quality, low-skew mesh.

Smoothing involves the movement of nodes in a face or volume to keep them evenly spaced and maintain low cell skewness. The remeshing method is activated when the skewness or cell size goes outside set limits. When this happens that section of the mesh is re-meshed with new nodes and cells and the information from the previous mesh cells is interpolated to the new mesh cells. Re-meshing takes more computational effort due to the checks and the computational requirements of calculating a new mesh, however it is necessary as the minimum and the maximum volumes of the expansion chambers vary significantly, as does the aspect ratio of the volume to surface area. Therefore, if remeshing did not occur, large changes in mesh density and high mesh skew values would occur causing inaccuracies or failures.

To maintain the mesh densities at the desired locations, the nodes and cells of the outline walls must be kept at the correct size. This is done by ensuring that the initial mesh design has a very good quality with low skewness plus the correct cell size for the corresponding locations. Having the correct cell size on the boundary walls, which do not experience automatic smoothing or

re-meshing, forces the internal mesh near that area to have a similar cell size to maintain mesh quality, thereby ensuring correct mesh density.

4.5.5. Fluent Setup

When it comes to selecting the various controls in the setup of the Fluent simulation, most of the controls can be set the same regardless of the different conditions being modelled. The first option that needs to be chosen is the solver type. Following the advice from the Ansys help manual [109], it does not matter much which of the initial two solver types is chosen, however it mentions the density-based solver is better for simulations with high speed compressible flow, which should not occur in these simulations. Therefore, the pressure-based solver was chosen. The pressure-based solver has two main types of solution algorithms, segregated, and coupled. The segregated type solves the governing equations faster and uses less memory. However, as it must solve the equations sequentially, the convergence is not as good as the coupled algorithm. As the geometry being simulated in this case has both complex shapes and complex motion, convergence is much harder to achieve. Therefore, the coupled algorithm was chosen to give the solution the best chance of convergence at each time step.

The last controls related to the solver are the gradient and under relaxation factors, these were chosen through trial and error to find the options that produced the fastest converged solutions. Table 4.2 shows the values used in the simulations.

Table 4.2 – Simulation controls used in Fluent models.

Control		**Value**
Solver type		Pressure based
Solver time		Transient
Solution numerical coupling scheme		Coupled
Spatial Discretization:	Gradient	Least squares cell based
	Pressure	Second order
	Density	Second order upwind
	Momentum	Second order upwind
	Turbulent kinetic energy	First order upwind
	Turbulent dissipation rate	First order upwind
	Energy	Second order upwind
Transient formulation		First order implicit
High order term relaxation?		Yes
Flow courant number		10
Explicit relaxation factors:	Momentum	0.5
	Pressure	0.5
Under-relaxation factors:	Density	0.5
	Body forces	0.5
	Turbulent kinetic energy	0.4
	Turbulent dissipation rate	0.4
	Turbulent viscosity	0.5
	Energy	0.5

Boundary conditions had to be defined next, the types of boundary conditions used in these models are wall, interface, pressure inlet and pressure outlet. The wall boundary conditions are defined as no-slip and with zero heat transfer (adiabatic). The no-slip condition is used to reduce computational effort as near wall behaviour is not of large importance in the studies carried out in this research. The zero heat flux condition is chosen as the heat transfer properties of the solid materials are not known and neither are the properties and conditions of the outside environment. Further investigation later in the development process will be required to determine an accurate heat transfer for the model. The interface boundary conditions are applied to all the faces in which fluid can pass between one section of the mesh and another. These interface faces act as walls when there is no other interface mating with it but allow free flow when there is. The pressure inlet and outlet boundary

conditions essentially achieve the same purpose. They attempt to keep the defined face at the specified pressure, adjusting the flow rate in/out accordingly.

One of the final sections to setup before starting a simulation is defining the output files. The simulation is setup to save a results file every 10 time-steps, this is not done every time step as the results files take up a large amount of space on a computer hard drive. Additionally, files are written at every time-step containing the average pressure values for each expansion chamber, the volume of each expansion chamber and the total mass flow inlet. These output into comma separated values (CSV) files which can then be read and processed in a variety of different ways. These files are much smaller in size and therefore can include results for all time steps.

As the simulations are run in transient mode, the simulation must be split into time-steps and a solution determined at each one. For the initial design of the simulation model, trial and error was utilised to find the best time step size for the simulation. Reducing the time step has the disadvantage of increasing simulation time, therefore the ideal time step size would be the largest that still has convergence and does not result in other errors. Two reasons for error can results from a time step that is too large. The first is the simulation solution itself, if the change in flow properties from one time-step to the next is excessive, it can produce errors in continuity. The second is due to the dynamic mesh movement. If the mesh boundaries move too far in one step, they can pass the nearby interior mesh cells, causing those cells to be

outside of the boundary and resulting in 'negative cell volume'. Ultimately, the first of these concerns was never an issue, because for the speed of the expander, the mesh required a far smaller time step than would ever cause problems with the flow solution. The final decision on time step size was dependent on the rotational speed of the simulation. Through trial and error, it was found that splitting one full rotation into 400 time-steps generally prevented errors and produced an accurate solution. Therefore, the actual time step size varied according to the time taken for a full revolution.

The average time to complete a simulation was 200 hours for a 3D simulation without apex seals, 13 hours for a 3D simulation with apex seals and 6.5 hours for a pseudo-2D (section) simulation without apex seals. This was using a computer with Intel Core i7-4820k (3.7GHz) processors and 48GB of RAM running Windows 10 Enterprise 64-bit.

4.5.6. Results Post-Processing

Once the simulation is complete the results can be viewed and processed in two different ways. The first is using the Ansys CFD-Post software, which can read the Fluent results files produced every 10 time-steps. Here visual representations of the fluid flow can be produced using pressure/temperature contours, velocity vectors or streamlines. The contours and velocity vectors can be shown on any 2D plane or mesh face and the streamlines can be seen in 3D space. These can all be viewed as either still images or animations. This method of viewing the results is very useful for troubleshooting and

discovering reasons for different flow phenomena/abnormalities in trends, however the processing in this method takes a lot of time as the program has to calculate from the entire mesh results and from multiple files (from the different time steps). It is therefore not easy to compare many results.

The CSV file results files mentioned give a much easier and quicker way to process results. As they contain all the results, already averaged, in the same file, they are much faster to process. A MATLAB code was written to access all the files required, load the results into the software, and produce the required results outputs of power output and average inlet flow rate. The work done per cycle is calculated from the integration of a pressure-volume (PV) closed curve, created from volume-averaged pressure results of each chamber, an example of a PV diagram is shown in Figure 4.18. This is then used with the rotational speed to calculate the power output as shown in Equation 4.8. The final numerical results are stored in a Microsoft Excel file where the isentropic efficiency is calculated. This is achieved by calculating the specific work output of the expander and then dividing it by the ideal isentropic specific work output for the same pressure difference, as demonstrated in Equations 4.9 and 4.10.

$$P_{CFD} = 2W_{pv}\omega \tag{4.8}$$

Where P_{CFD} is the CFD power output, W_{pv} is the work calculated from a chamber's pressure-volume diagram per cycle and ω is the rotational speed in revolutions per second.

$$w_{CFD} = \frac{P_{CFD}}{\dot{m}_{inlets}} \tag{4.9}$$

$$\eta_{CFD} = \frac{w_{CFD}}{w_{ideal}} \tag{4.10}$$

Where w_{CFD} is the CFD specific work, $\dot{m}_{inlets}$ is the total cycle averaged mass flow rate for all inlets, η_{CFD} is the isentropic efficiency of the expander in CFD and w_{ideal} is the ideal specific work if the expander had 100% isentropic efficiency.

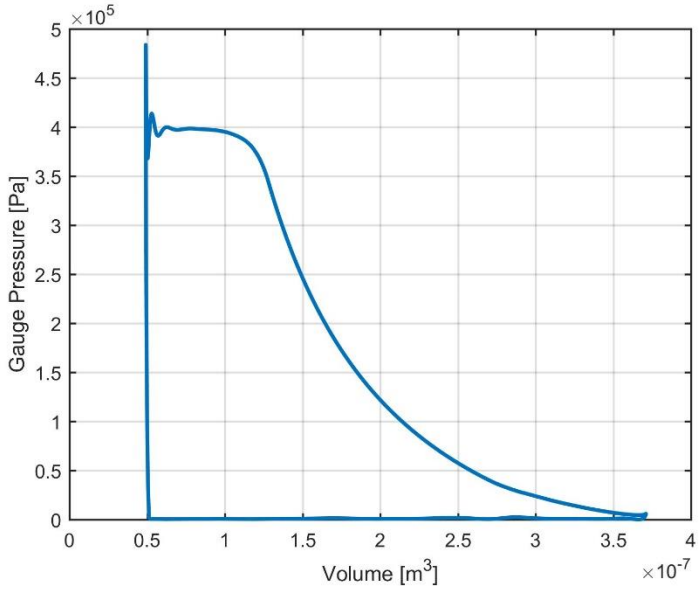


Figure 4.18 - Example pressure-volume diagram

4.6. Pseudo-2D CFD

Although 2D CFD is much quicker it is not entirely suitable for accurate modelling of the Wankel expander. This is mainly due to the inlet and outlet port arrangements. If the ports are side located then it is not possible to simulate in 2D, where no flow on the third dimension will be allowed. Therefore, for completely reliable and accurate results 3D CFD simulations

are required. However, as mentioned earlier, a full 3D simulation takes a lot of time and would make a full parametric study unfeasibly long. To speed up the process, the preliminary study was completed using a thin section of the mesh. This setup is similar to a 2D simulation in that it can be multiplied by the width to find an approximation of the performance. In many places it is also only one cell wide, which is the same as 2D mesh setups. This is made possible using 'Symmetry' wall boundary conditions, which exert no wall effects on the fluid in the same plane. However, it does differ to a 2-D simulation in some essential ways, mainly that it does allow flow in the z-axis in some places. This is important so that the side mounted outlet ports can be used.

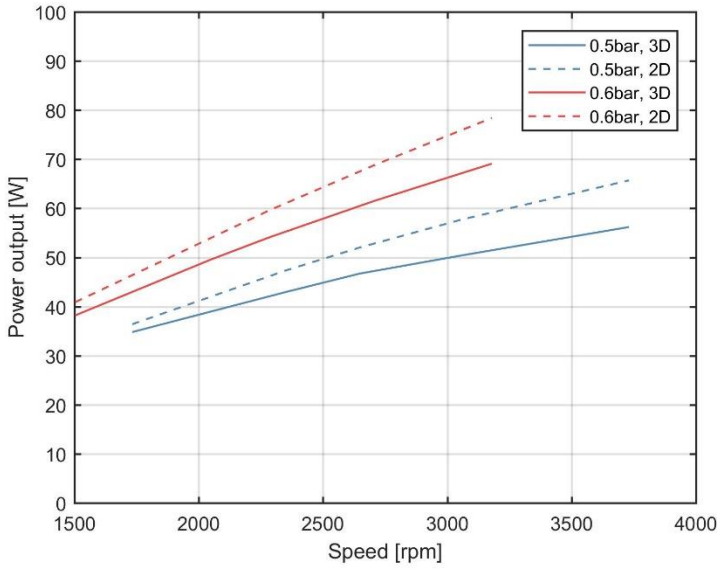


Figure 4.19 – Power output comparison between 3D and pseudo-2D simulations

This pseudo-2D setup will introduce new inaccuracies, as the simulation no longer represents the exact setup of the expander. One reason for the inaccuracies is the ratio of the expansion chamber volume to side port area is

changed. It is therefore much easier to fill and empty the chamber through these side ports in a pseudo-2D model. Furthermore, any effects on performance from z-axis flow regimes that can form will now be neglected. Figures 4.19 and 4.20 show the difference in simulation outputs between the pseudo-2D and 3D simulations, with the pseudo-2D results having an average deviation of 5.73W in power output and 1.74% in efficiency compared to the 3D simulation. However, the time savings make this an attractive option to find the range of operating conditions where optimal performance occurs. These solutions can be used to guide conditions used for the computationally heavy 3D simulations which can be used to find the accurate results. This is very useful as a pseudo-2D simulation generally takes 6-8 hours to complete whereas a 3D simulation can take between 168-432 hours to complete.

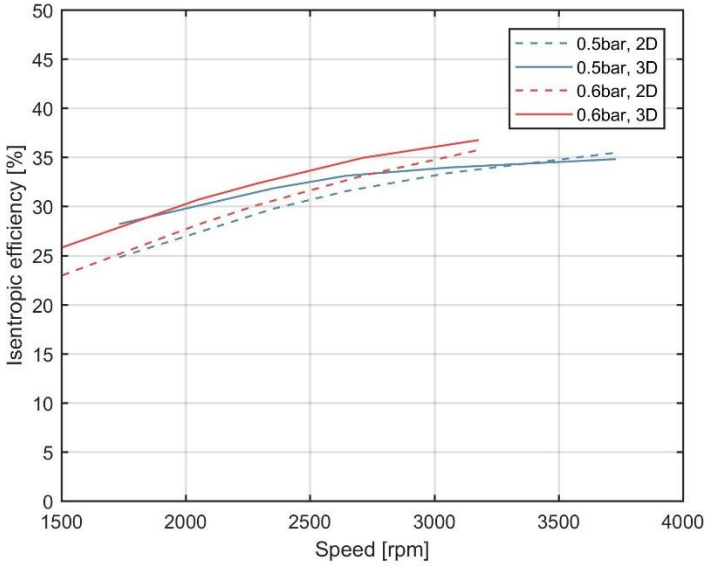


Figure 4.20 – Efficiency comparison between 3D and pseudo-2D simulations

4.7. Parametric Study

Once the CFD models were created and tested, they were utilised to find both the optimum operating conditions and find the effect of different apex clearances. To find optimum conditions with the multiple variables, a parametric study was set up. To keep the time down for the analysis, only one size of expander was considered (Radius = 30mm, Eccentricity = 4.125mm, Width = 20mm). This size was chosen based on the same R/e ratio as a previous experimental prototype, which had specific requirements due to gear sizes and centre spacing. Three different inlet pressures (2bar, 4bar and 6bar gauge pressure) were chosen to be used in the parametric study, the middle pressure (4bar gauge pressure) was chosen so that if the cycle was ideal, with the expander's volumetric displacement, the outlet pressure would be approximately ambient (0bar gauge pressure).

The rotational speed was also varied, with speeds of 4800, 6000, 7200 and 8400rpm chosen for the static shaft Wankel expander and 1200, 3000, 4800, 6000rpm for the standard Wankel expander. The only geometric parameter that was varied was the apex clearance. Originally more apex clearances were analysed, including 0.5mm, 0.25mm, 0.1mm and 0.05mm and a fully sealed (0mm). However, analysis of initial CFD results found that the 0.5mm and 0.25mm apex clearances did not produce any viable results.

4.8. Apex Seal Friction Estimation

To compare the cases with and without apex seals fairly, the friction losses from the apex seals should be accounted for. To achieve this, Equations 4.11, 4.12 and 4.13 were utilised. These equations use the spring forces, the normal forces due to pressure and the centripetal forces on the seals to estimate the power loss due to friction. These equations do not account for the change in contact angle of the apex seals during a cycle, however, this change was considered minimal for a thin section seal and acceptable to ignore. The coefficient of friction, μ was taken as 0.16, which is an average for steel on steel with oil lubrication.

$$f_{seal,centripetal} = m_{seal} r_{COM} \left(\frac{4\pi\omega}{3}\right)^2 \quad (4.11)$$

Where $f_{seal,centripetal}$ is the normal centripetal force between the seal and housing, m_{seal} is the mass of the seal in kg, r_{COM} is the radius of the circular path the seal's centre of mass travels, and ω is the rotational speed in rev/s.

$$f_{seal,pressure} = p_{Average} A \quad (4.12)$$

Where $f_{seal,pressure}$ is the normal pressure force on the seal, $p_{Average}$ is the cycle average pressure difference between two chambers, and A is the area of underside of a seal.

$$P_{friction} = 3\omega\mu r_{contact}\left(f_{seal,centripetal} + f_{seal,pressure} + f_{seal,spring}\right) \quad (4.13)$$

Where $P_{friction}$ is the cycle averaged power loss due to all apex seal friction, μ is the coefficient of friction between the seal and housing, $r_{contact}$ is the mean

distance the apex seal to housing contact occurs, and $f_{seal,spring}$ is the normal force on the apex seal due to the leaf spring, which is located behind each seal.

4.9. Chapter Summary

- The Wankel expander without apex seals was described with its possible benefits.
- The static shaft Wankel expander was introduced and described, showing the advantages of integrated inlet timing control and intrinsically balanced rotary parts.
- The Wankel expander CFD model design and setup was described, including mesh sensitivity studies which showed a maximum variation in efficiency results of 0.558% due to mesh size.
- The UDFs required for the transient Wankel expander motion were described.
- The method for analysing and post-processing the CFD results was given.
- A pseudo-2D CFD model was described to reduce computational time in parametric studies.
- A method for estimating apex seal friction was presented, which can be used to help compare the CFD models of the Wankel expanders with and without apex seals.

CHAPTER 5

WANKEL EXPANDER CFD SIMULATIONS: RESULTS

5.1. Chapter Introduction

The first half of this chapter looks at the CFD results of an analysis of two modifications to the Wankel expander using compressed air as the working fluid. The first modification was designing the expander without any apex seals and the second was using a new design to achieve a similar performance to a setup with inlet valves, but without the previously mentioned disadvantages. The analysis models a variety of design variations and operating conditions to save on the cost of producing and testing multiple prototypes.

Another application where the developed small-scale static shaft Wankel expander may show benefits is in small scale organic Rankine cycle systems. These systems were described in the literature review chapter (Section 2.5) and involve four main components: a pump, an evaporator, an expander and a condenser. Of these, it was widely considered that the expander performance had the largest effect on the cycle performance [5, 6]. The second half of the chapter describes a study into how the static shaft Wankel expander performs using two frequently used refrigerants in the organic Rankine cycle.

5.2. CFD Results: Standard Wankel Expander

5.2.1. Apex Seal Friction

Figure 5.1 shows a comparison of some results with and without the estimated apex seal friction included. Including the friction, the expander's efficiency is reduced by a maximum of 0.73% and power output by a maximum of 3.9W. It was therefore decided that for the purposes of this analysis the friction could be omitted without significant consequences. The main disadvantage of apex seals, for the case of the standard expander, would therefore be the added complexity and need for a lubricating oil system.

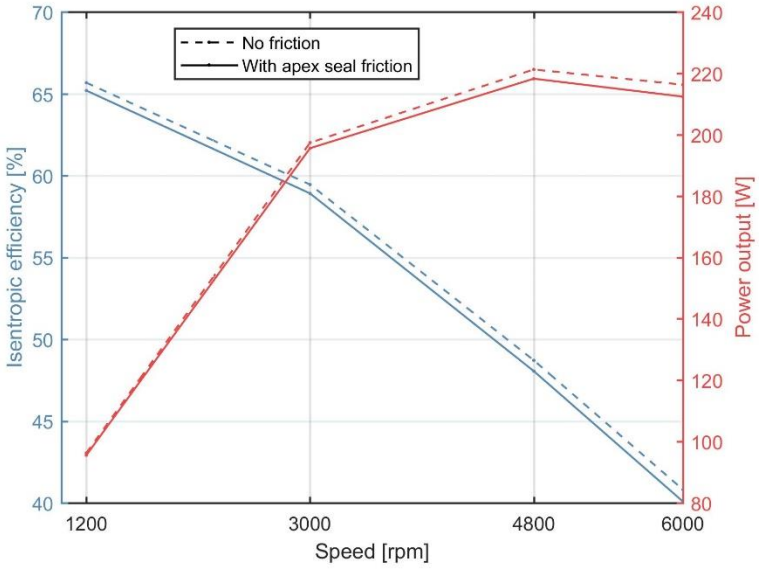


Figure 5.1 –Comparison of cases with and without estimated apex seal friction (2barg inlet pressure)

5.2.2. Rotational Speed & Double-Sided Ports

Initial results from the CFD of the standard Wankel expander without external valves (with the port timing relying on the rotor position) showed that there were some issues with high-speed operation. Figure 5.1 shows how

the efficiency of the standard expander with apex seals varies with speed. The efficiency is reasonable at the low speed of 1200rpm. However, for the higher speed cases the efficiency drops dramatically. These results led to further analysis of the individual CFD results, where it was found that there were large pressure drops over the inlet and outlet ports due to their relatively small size. The side inlet ports cannot be simply enlarged as this would alter the opening timings causing increased under-expansion.

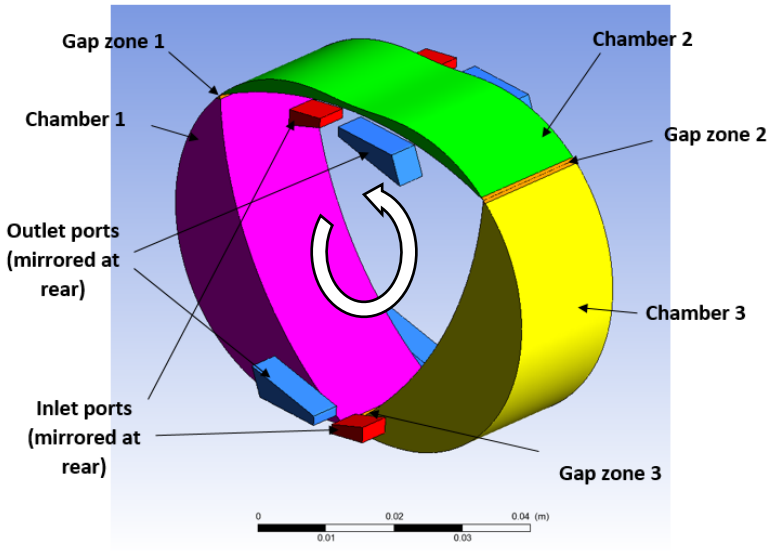


Figure 5.2 – Geometry of the double-sided port setup of the standard Wankel expander

To fix this issue and allow the effect of apex clearances on performance to be fairly compared, additional ports were added, so that there are inlet and outlet ports symmetrically on both sides of the expander as in Figure 5.2. This setup is called the 'double-sided ports' setup. The same simulation parameters as before were used but with this new setup and the isentropic efficiency results are compared in Figure 5.3. The case with apex seals

benefits greatly from the double-sided ports setup with a much-reduced drop in efficiency at high speed. The double-sided ports setup also allows the 0.05mm clearance to reach a much higher efficiency. However, there is still a limit to the efficiency which can still be seen to peak and start to drop at the higher speed cases.

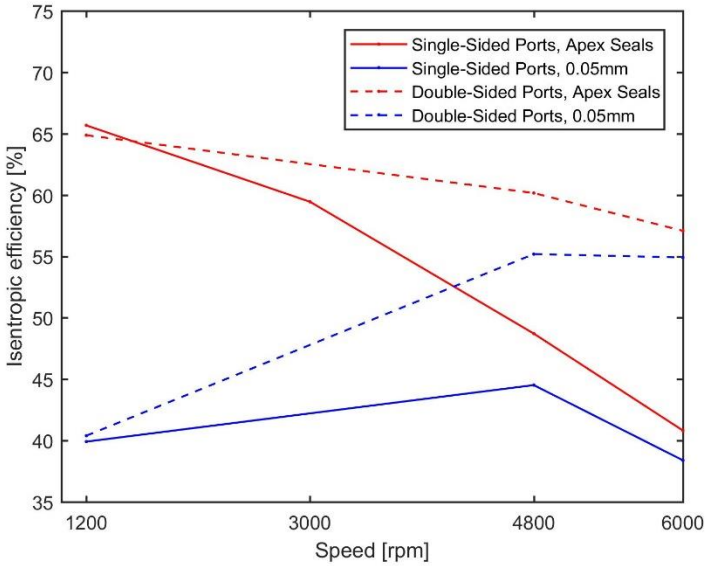


Figure 5.3 – Comparison of isentropic efficiency against speed for single-sided and double-sided port setups (2barg inlet pressure)

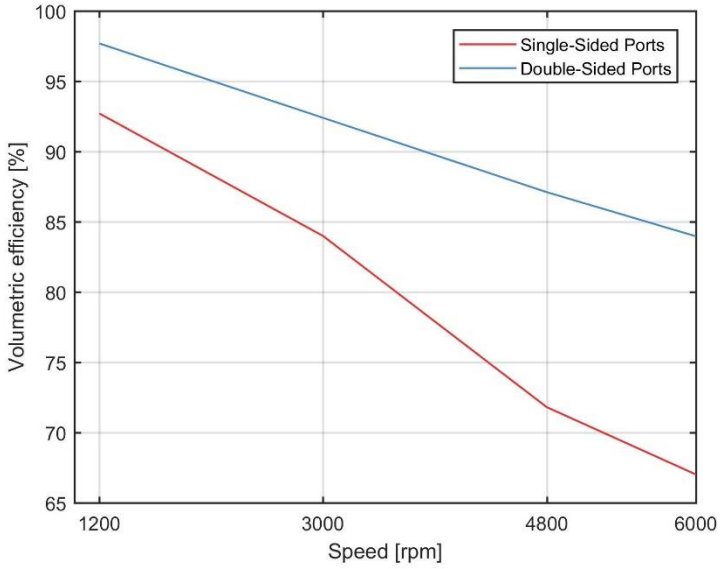


Figure 5.4 – Comparison of volumetric efficiency for single-sided and double-sided port setups (2barg inlet pressure, with apex seals)

Another way to analyse the losses due to the ports is volumetric efficiency which is shown in Figure 5.4. Volumetric efficiency is defined as the actual over the ideal inlet flow rate. Cases without apex seals are not included as they cannot be fairly compared using volumetric efficiency because the increased internal leakage will increase inlet flow rate without giving any performance benefit, making the results misleading. The results with apex seals explain the isentropic efficiency trend. In both cases the volumetric efficiency is always decreasing with speed, however, the double-sided ports cases decrease at a lower rate. Therefore, it can be concluded that an increase in rotational speed results in a reduced leakage through the apex clearances but also results in a decreased volumetric efficiency. Depending on the extent of each effect, an optimal rotational speed will exist for the best isentropic efficiency. The highest isentropic efficiency for the double port setup was only 65%, which had a low power output, as shown in Figure 5.5. As no other way

to increase the volumetric efficiency could be found, different design options which can increase it further were studied.

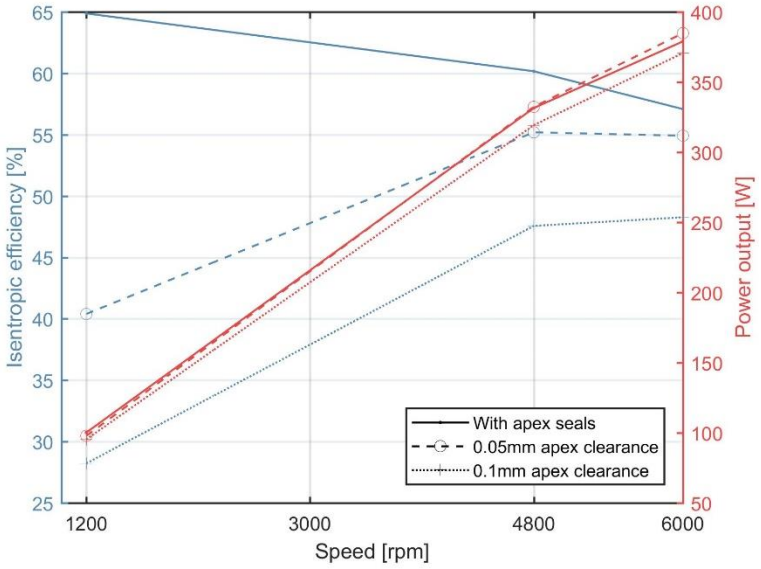


Figure 5.5 – Efficiency and power output against speed for different apex clearances (2barg inlet pressure, double-sided ports)

5.2.3. Inlet Pressure

Figure 5.6 shows how the isentropic efficiency and power output vary with inlet pressure. Here it demonstrates the major disadvantage of the standard Wankel expander with no inlet valves. Relying on the ports for inlet timing means the inlet port is open for a large portion of the expansion phase, making most of the expansion isobaric. The time that the inlet is open cannot be reduced due to the geometry and motion of the rotor, which provides the opening and closing mechanism. Higher inlet pressure results in increasingly larger under-expansion losses and lower efficiency but low inlet pressure leads to a low power output.

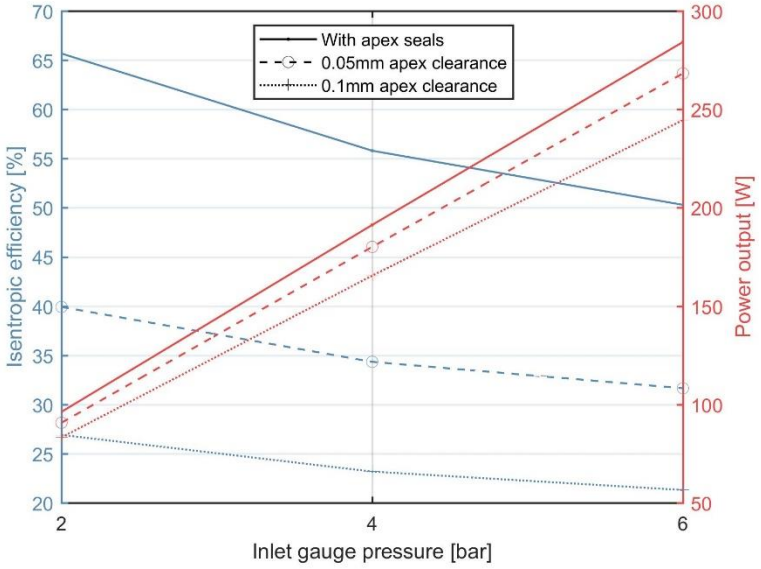


Figure 5.6 – Efficiency and power output against inlet pressure for different apex clearances (1200rpm, single-sided ports)

The maximum efficiency of all standard Wankel expander cases is 64.88%, which does not compete with other leading expanders on the market. One solution to increase efficiency would be to introduce a valve system on the inlet ports, however as this would increase complexity, cost, and introduce new losses to the system, it would not be the ideal solution. Another possible solution is using the static shaft expander setup, the results of which are looked at later.

5.2.4. Apex Clearances

Figure 5.7 shows the effect that apex clearance has on the efficiency of the expander. The efficiency of the 0.25mm apex clearance is too low and therefore is not considered feasible under any conditions. The 0.1mm and the 0.05mm apex clearances approach more acceptable efficiencies at high-speed. The isentropic efficiency of the 0.05mm apex clearance case at 6000rpm is

similar to the efficiency of the cases with apex seals, suggesting the seals do not add much benefit at this high-speed. Furthermore, the 4800 and 6000rpm cases show very little difference in efficiency, which suggests that further speed increase would have little benefit for efficiency. However, here it is hard to ascertain whether the primary influence on the efficiency is the apex clearance or the inlet port pressure drop effects.

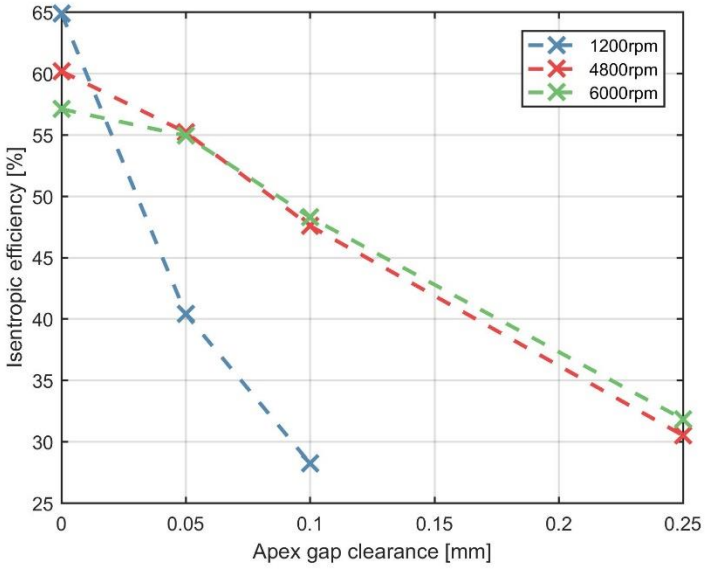


Figure 5.7 – Expander efficiency against apex clearance for different rotational speeds (2barg inlet pressure, double-sided ports)

5.3. CFD Results: Static Shaft Wankel Expander

5.3.1. Apex Seal Friction

Similar to the previous design, Figure 5.8 shows that including apex seal friction estimations has little effect on the output values. Therefore, the apex seal friction was omitted from the following simulations as well.

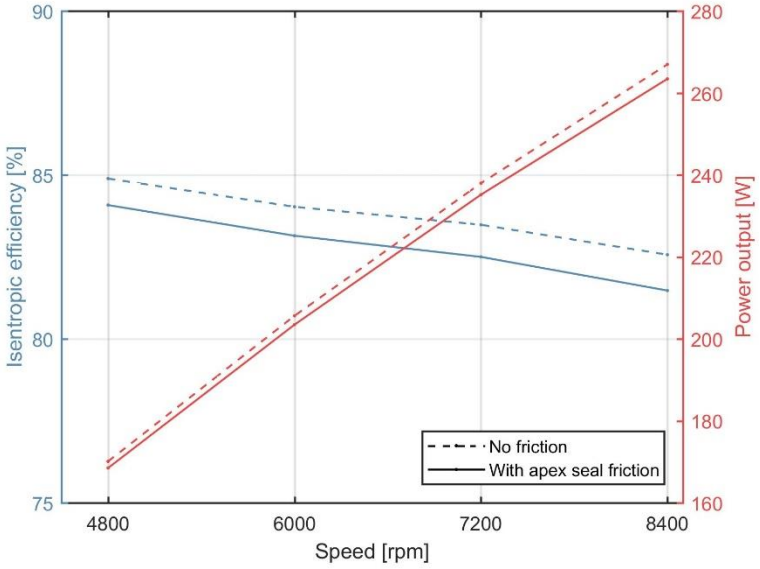


Figure 5.8 – Comparison of cases with and without apex seal friction (2barg inlet pressure, sealed cases, best inlet timing)

5.3.2. Rotational Speed

The effect of speed was analysed to find if the new design had reduced the inlet port pressure drop problems seen in the previous design. Figure 5.9 shows isentropic efficiency results against speed. For the cases with apex seals, there is almost no change in efficiency over the speed range analysed. This is an indicator that any pressure-drop over the inlet port has a negligible effect on the performance. Furthermore, the cases with 0.05mm and 0.1mm apex clearances have continuously increasing trends, suggesting the same as there is no peak established within the speed range.

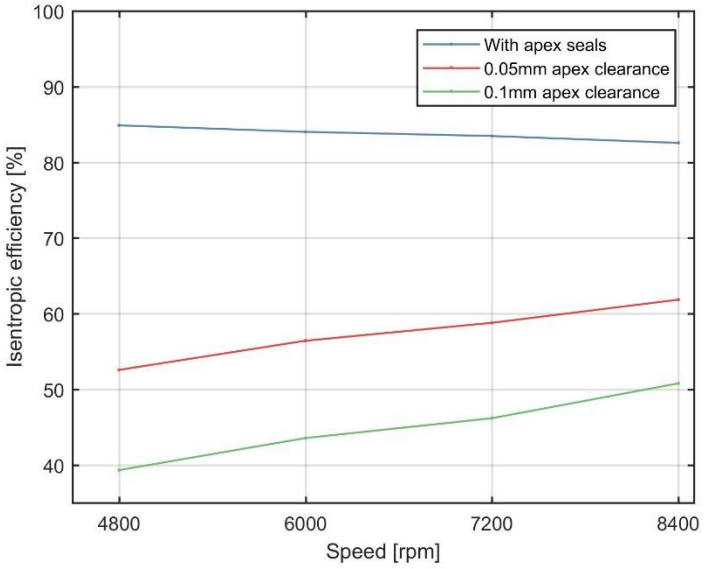


Figure 5.9 – Efficiency against rotational speed for different setups (2barg, optimal inlet timing)

Figure 5.10 shows a comparison of volumetric efficiency for the previous standard Wankel expander design and the new static shaft Wankel expander. For the same reason as before, only cases with apex seals are shown. The static shaft Wankel expander cases were modelled at a higher rotational speed range, because it wasn't limited by the pressure drop effects as the standard Wankel expander was. The volumetric efficiency for the static shaft Wankel expander is only slightly better than the standard Wankel expander at 4800rpm, however, the volumetric efficiency remains high over the entire speed range, with a much higher maximum speed. This more viable high-speed operation helps reduce apex clearance leakage which is the reason for higher isentropic efficiency values seen in Figure 5.9.

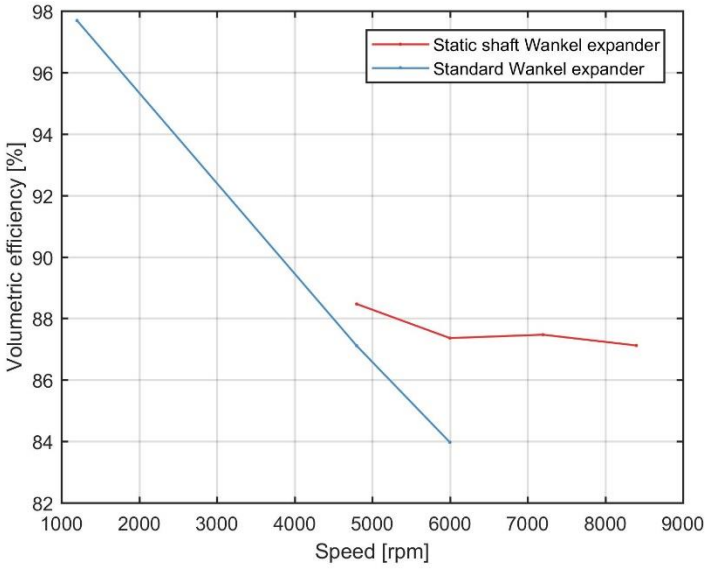


Figure 5.10 – Volumetric efficiency against speed for standard and static shaft Wankel expanders (with apex seals, optimal inlet pressures)

5.3.3. Inlet Timing

This section looks at how the performance of the static shaft Wankel expander is affected by the inlet opening timing, which can now be precisely chosen. For a rotational speed of 4800rpm, Figures 5.11 and 5.12 show the effect of inlet opening time for the cases with apex seals and with 0.05mm apex clearances respectively. For the case with apex seals, the 2barg inlet pressure has a continuously increasing efficiency over the range, this suggests that the gas is experiencing over-expansion which gets worse with shorter opening times. The 6barg inlet pressure shows continuously decreasing efficiency, this conversely suggests that under-expansion is occurring and gets worse with longer opening times. 4barg inlet pressure had the best performance, with a peak at the mid inlet timing value of 60 degrees. The cases with 0.05mm clearances suffer less from longer opening times and the 4barg and 6barg case

now share a very similar curve, however, the 60deg inlet timing still performs best for all inlet pressures.

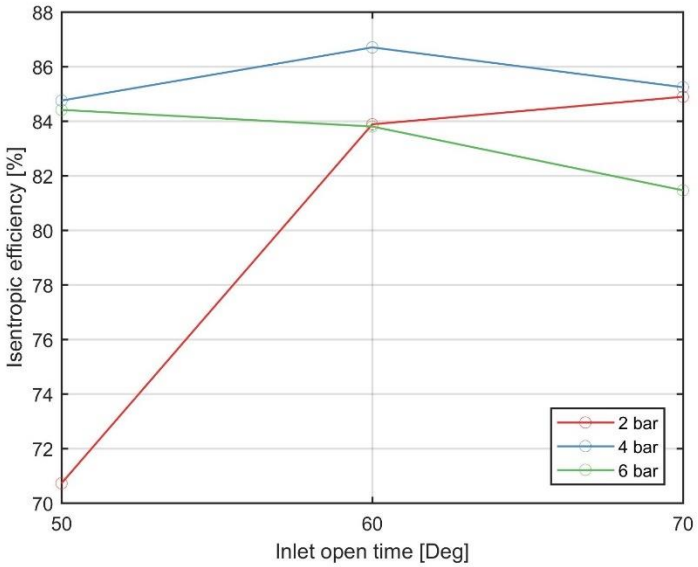


Figure 5.11 – Efficiency against inlet opening timing for different inlet gauge pressures (with apex seals)

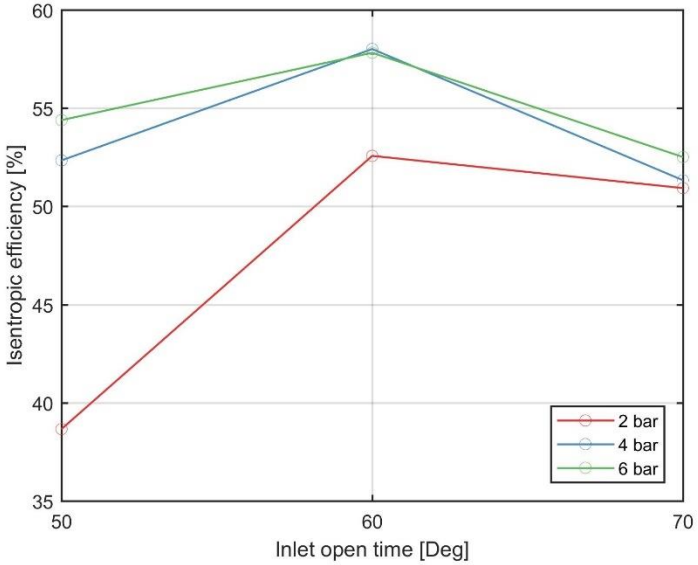


Figure 5.12 - Efficiency against inlet opening timing for different inlet gauge pressures (with 0.05mm apex clearances)

5.3.4. Inlet Pressure

An example of how the in-chamber pressure varies with time, over one cycle, can be seen in Figure 5.13. The near vertical line of pressure increase shows there is minimal restriction during the gas inlet and the near zero pressure shows a small pressure drop during the exhaust. When analysing the effect of changing inlet pressures, it is shown in Figure 5.14 that for the cases with apex seals, 6barg inlet pressure causes a decrease in efficiency due to under-expansion. The cases without apex seals do not suffer such efficiency loss at higher inlet pressure and have similar efficiency values for both 4barg and 6barg. Therefore, the cases without apex seals require a higher inlet pressure than the case with apex seals to attain their optimal efficiency.

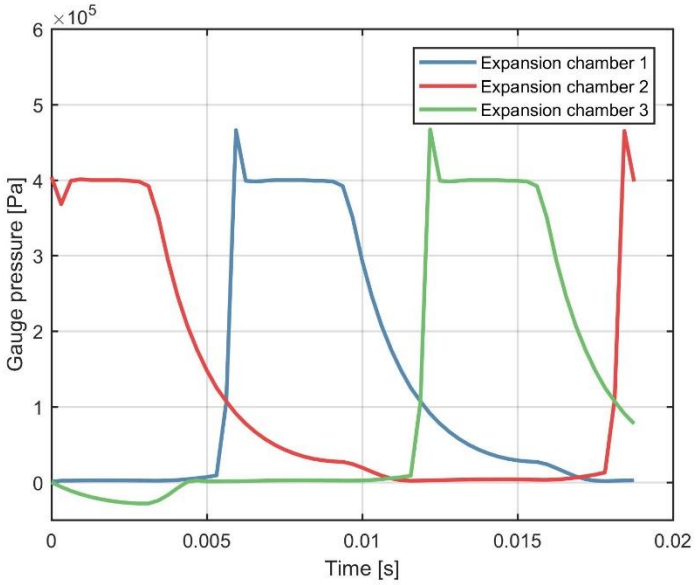


Figure 5.13 – Chamber pressure against time for 1 cycle (4800rpm, 4barg inlet pressure)

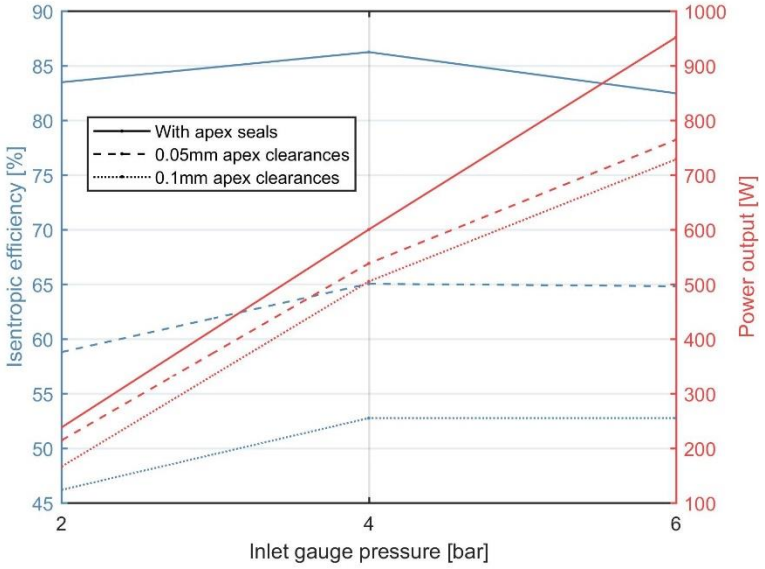


Figure 5.14 – Efficiency against inlet pressure for different setups (7200rpm, 60deg inlet timing)

The power output increases almost linearly for the case with apex seals and even the cases without apex seals display an always increasing power output over the pressure range. Therefore, for maximum power the pressure should be as high as possible, providing the efficiency remains reasonable and the device is designed to handle the pressure.

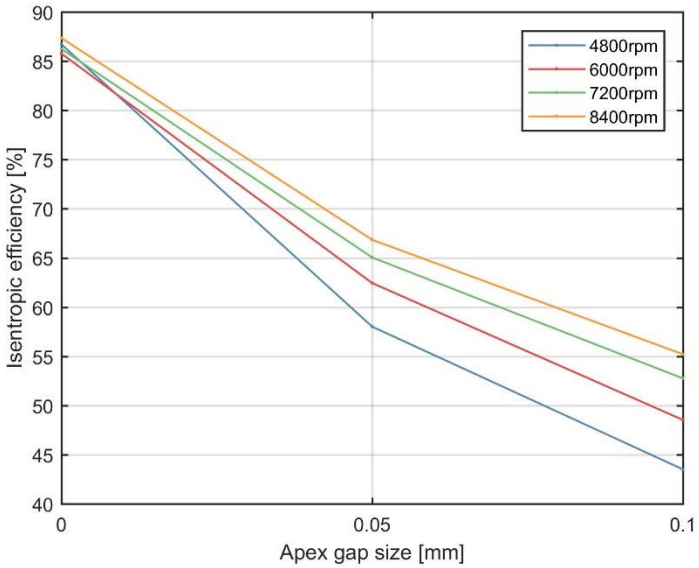


Figure 5.15 – Efficiency against apex clearance for different speeds (2barg inlet pressure)

5.3.5. Apex Clearances

Figure 5.15 demonstrates that the apex clearances have the largest impact on the performance of the expander. There are small leakage reductions with increased speed, noticed up to 6000rpm in the 0.1mm clearance case, however, further speed increase appears to have little impact on the performance of the expander. Figures 5.16, 5.17 and 5.18 show how the apex clearance and the speed affect the area of the pressure-volume diagram, which has a direct effect on the power output of the expander. It is noticed that, although there is a definite reduction in area, it is not a large amount. Therefore, it is assumed that the major reason for efficiency loss comes from increased inlet flow rate due to leakage.

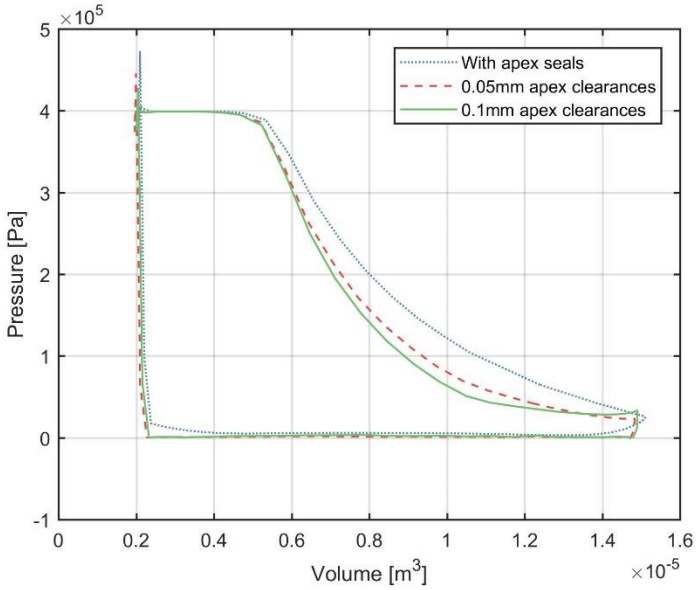


Figure 5.16 – Pressure-Volume diagram of static shaft expander with different apex clearances (6000rpm, 4barg inlet pressure)

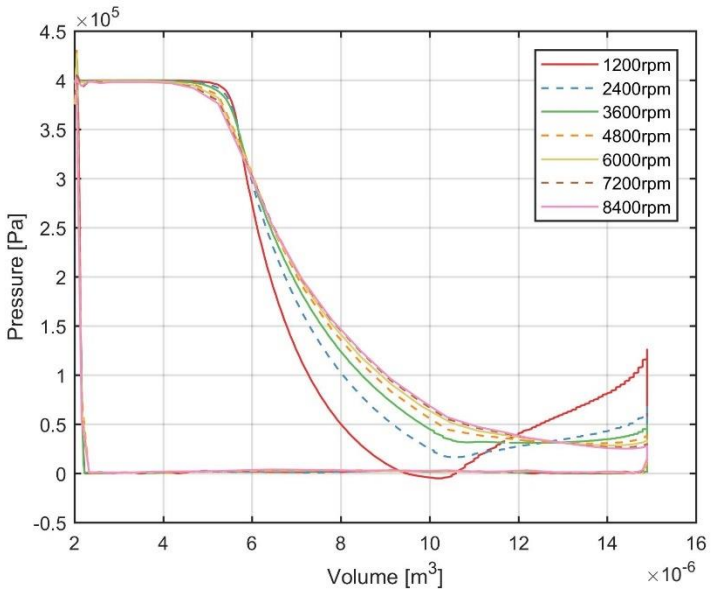


Figure 5.17 – Pressure – Volume diagram of the static shaft Wankel expander at different rotational speeds (0.1mm apex clearance, 4barg inlet pressure)

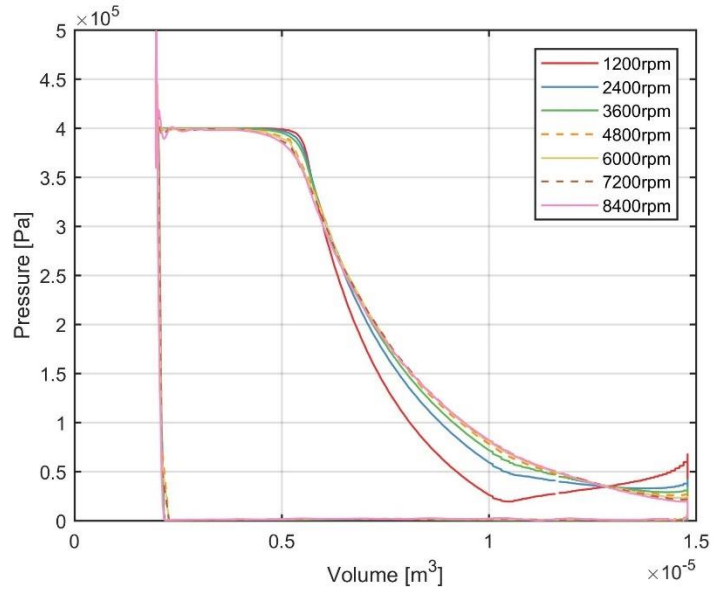


Figure 5.18 – Pressure-Volume diagrams of the static shaft Wankel expander at different rotational speeds (0.05mm apex clearance, 4barg inlet pressure)

Further investigation into the leakage flow through the apex clearances was required to better understand the leakage. Results were collected from the saved CFD files for the mass flow rates, velocities and temperatures at the apex clearances. The results in Figure 5.19 show that the Mach number of

apex leakage flow, which is the flow from one expansion chamber to another, is almost identical for all speeds and apex clearances (the negative values indicate reverse flow). Figure 5.20 shows how the mass flow rate varied with apex clearance but appears to be almost identical for different rotational speeds. This suggests that the rotational speed of the expander has no effect on the flow through the apex clearances, and it is simply the effect of less time for leakage to occur that improves efficiency. This is backed up by Figure 5.21 which shows the mass of fluid that passes the apex clearance per timestep (all cases have the same number of timesteps per revolution), this shows that even though the mass flow rates are the same, the total mass of gas leakage in a revolution will vary with rotational speed.

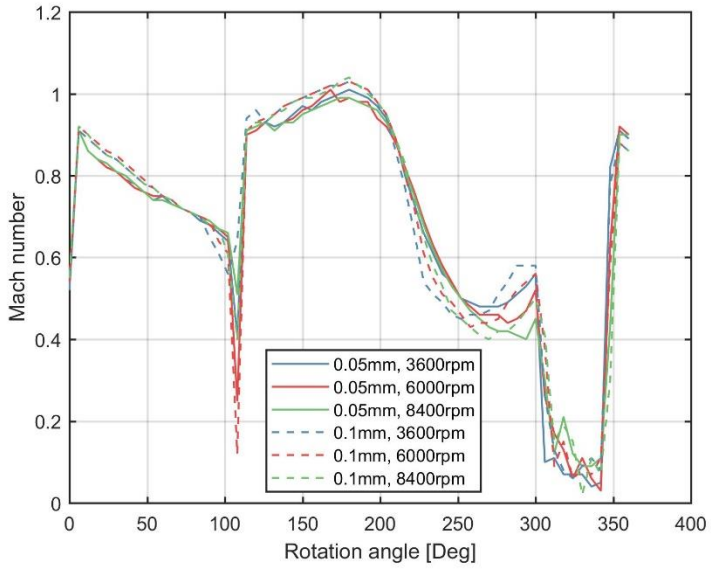


Figure 5.19 – Mach number of the flow past the apex for different clearances and rotational speeds

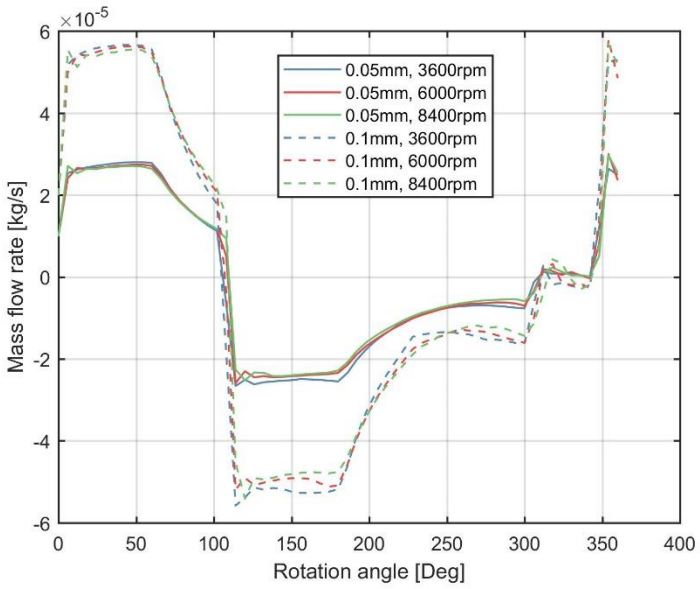


Figure 5.20 – Mass flow rates of the flow past the apex for different clearances and rotational speeds

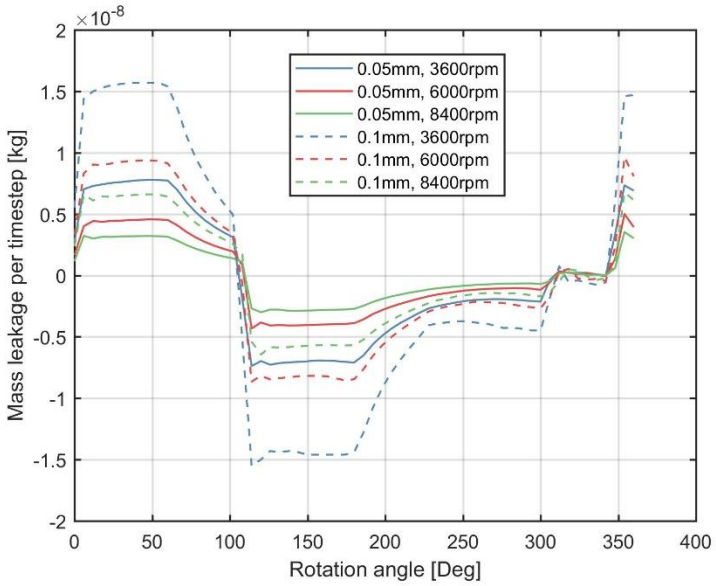


Figure 5.21 - Mass of fluid passing the apex per timestep for different apex clearances and rotational speeds

The pressure difference over the apex can be seen in Figure 5.22, where it is seen that the pressure difference between the two sides of an apex is very similar in all cases. This agrees with the fact that the pressure-volume diagrams shown earlier do not show much variation. If the pressure difference is the same for all cases, it can be inferred that the leakage through

the apex clearances is linearly proportional to both rotational speed and apex clearance. Therefore, the only way to reduce apex leakage, without changing the design to include oil flooding or apex seals, would be to either reduce the apex gap further or increase the speed further.

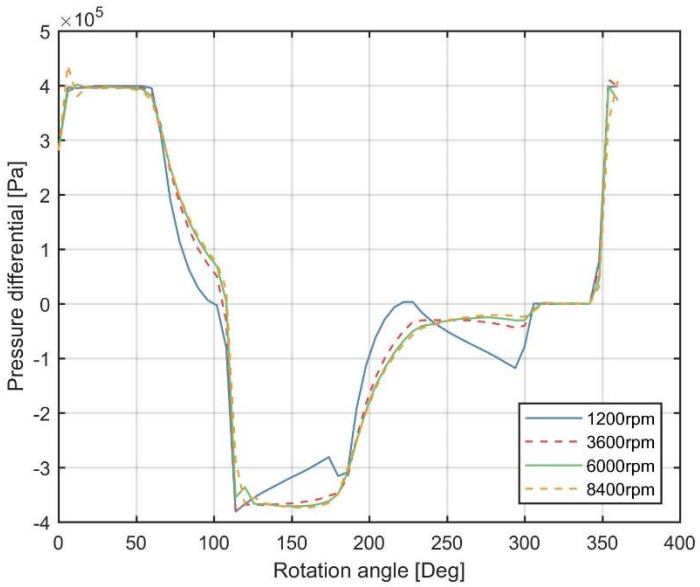


Figure 5.22 – Pressure differential over an apex with rotation angle for different rotational speeds (0.1mm apex clearance)

5.4. Comparing the Wankel Expanders

The CFD results for this static shaft Wankel expander compared with the previously analysed standard Wankel expander show its benefits. Figure 5.23 shows the efficiency at optimal operating conditions of both designs and it can be seen that the Static Shaft expander offers a large improvement. The case with apex seals offers an efficiency high enough to compete with other expanders on the market. The 0.05mm apex clearance case for the static shaft Wankel expander is so much improved that it outperforms the best case with apex seals of the standard Wankel expander.

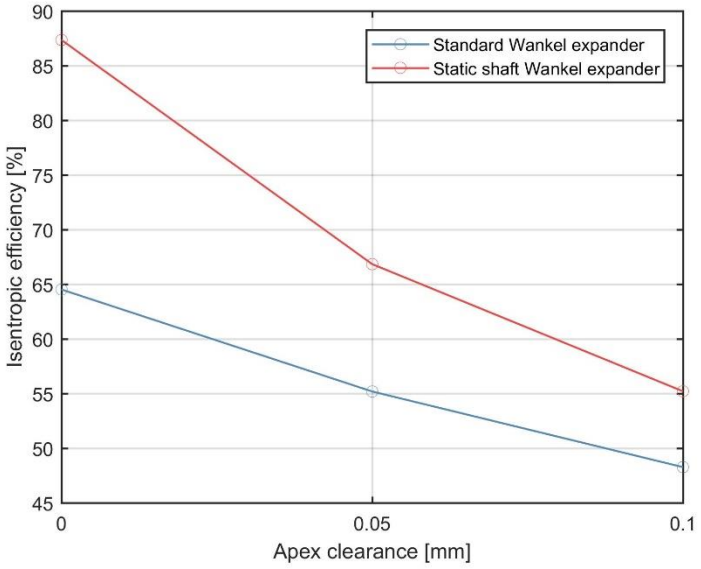


Figure 5.23 - Efficiency against apex clearance for original and static shaft expanders (optimal operating conditions)

Figure 5.24 shows the corresponding power output for the same cases and it is also much lower for the standard Wankel expander design. The power increases with apex clearance for the standard Wankel expander cases because the optimal conditions for efficiency have the speed increasing with apex clearance.

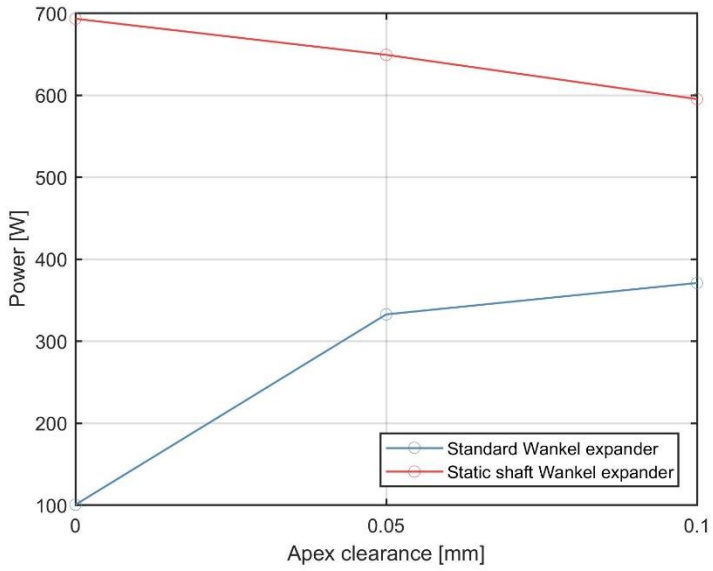


Figure 5.24 – Power against apex clearance for standard double-sided ports Wankel expander and static shaft Wankel expander (optimal efficiency operating conditions)

5.5. Comparing Expanders in Literature

The inlet to the static shaft Wankel expander acts as an inlet valve, controlling the opening and closing timing. Therefore, it is useful to compare its performance directly with the external valves used with the Wankel expander developed by Antonelli et al. [77], this is shown in Figure 5.25 where the calculated discharge coefficients are compared. The discharge coefficient was calculated as the actual flow (CFD flow) divided by the theoretical flow through an ideal orifice of constant area (equal to the area of the valve when fully open). It can be seen that the discharge coefficients are very similar, with the valve from Antonelli et al. having a higher peak, but the static shaft Wankel expander having a flatter peak, which will allow for higher flow over a larger period. The static shaft Wankel expander should therefore perform similarly to the expander developed by Antonelli et al. during the inlet phase,

with the added advantage that the 'valve' is included inside the expander body and produces less auxiliary power losses.

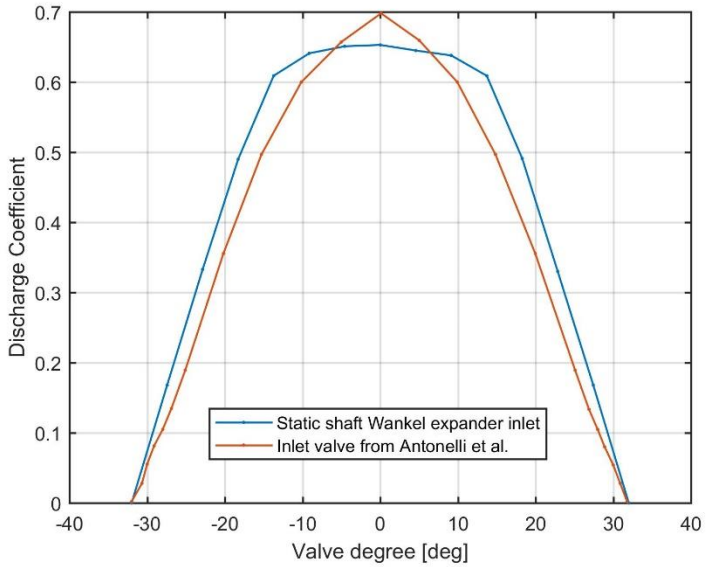


Figure 5.25 – Comparison of the inlet valve discharge coefficient of the static shaft Wankel expander and inlet valves from Antonelli et al. [77]

Figure 5.26 shows comparisons of the maximum isentropic efficiencies reported for various expansion devices in literature and the best results from the Wankel expansion devices studied in this research. The efficiency values reported are from numerical results and do not include additional losses from friction or leakage. Therefore, the results can be directly compared on a fair basis.

Although the static shaft Wankel expander has much improved performance over the original Wankel expander without valves, the case with valves reported by Antonelli et al. [77] still has a larger maximum efficiency. However, the static shaft Wankel expander in this study has further opportunity to receive optimisation. Other factors may also influence the

results, such as difference in size and working fluid (air/refrigerant/steam). The static shaft Wankel expander also has other advantages compared to the Wankel expander with valves, including greater simplicity, higher operating speed potential, smaller size, and lower weight.

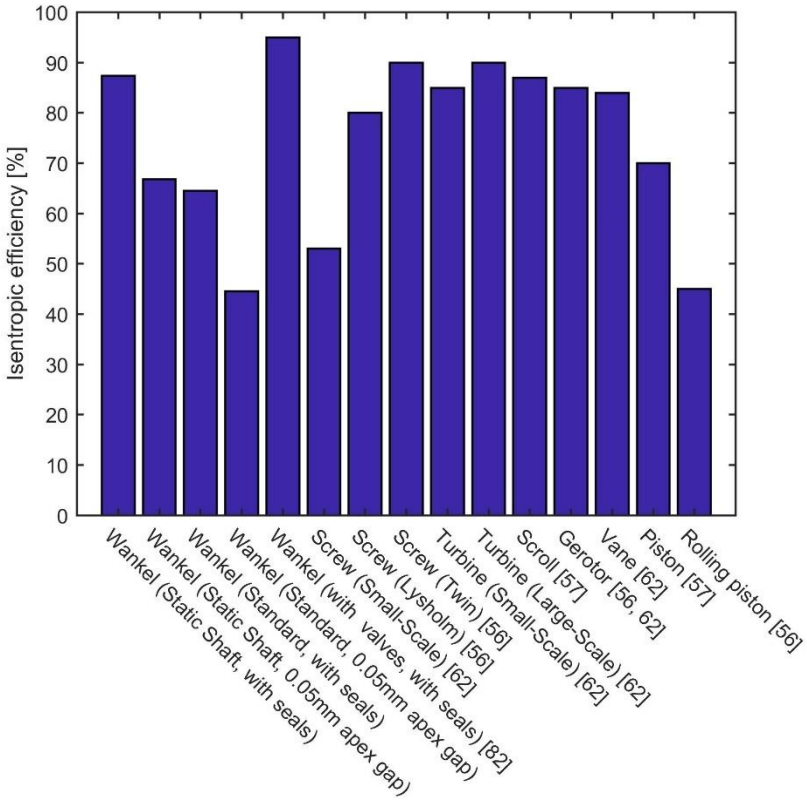


Figure 5.26 – Maximum reported efficiency values in literature for different types of gas expansion devices

5.6. Wankel Expander for ORC

As the Wankel expander developed in this project was designed for small-scale applications and offers much flexibility in terms of operating conditions, it also presents the opportunity to be used in an Organic Rankine Cycle (ORC) system. Its volumetric nature allows much lower flow rates to be utilised and its rotary motion means it has low noise and vibration, which would be good

for systems based near where humans live/work. Furthermore, as it has a simple construction, it would be relatively cheap to produce and easy for the user to assemble/disassemble for any maintenance.

Due to these reasons, the designed Static Shaft Wankel Expander was analysed using CFD to understand how the performance would differ under the specific conditions.

5.6.1. CFD Modelling Considerations

As the Wankel expander in the ORC cycle still operates as a gas expander, the same CFD model as in the previous air simulations can be used, with the exception of the following two key considerations.

The first consideration when altering the CFD model for ORC situations is the fluid itself. For the following simulations n-butane and n-pentane were utilised as they are both commonly used in ORC systems reported in literature and both are available as predefined fluids in Ansys Fluent 18.2. In previous simulations using air as the working fluid, the ideal gas model was utilised, as it offers the least computational effort and is sufficiently accurate at the investigated operating conditions. Due to the nature of the organic fluids, they are close to condensing conditions at the pressures and temperatures that occur in the CFD simulation. Therefore, the ideal gas model would not produce accurate results and the Peng-Robinson real gas model had to be used. Equations 5.1 to 5.5 show the equations used in the Peng-Robinson real gas model, from the Fluent help manual [109]. The

equation of state is simply used instead of the ideal gas equation, using the critical pressure critical temperature and acentric factor to define each different gas.

$$P = \frac{RT}{V - b} - \frac{\alpha}{(V^2 + 2bV - b^2)} \tag{5.1}$$

Is the main equation of state where P is the absolute pressure, R is the universal gas constant, T is the temperature, V is the specific molar volume, α is the attractive coefficient and b is a constant dependent on the gas's critical pressure and temperature.

$$\alpha = \alpha_0 \left[1 + n\left(1 - \left(\frac{T}{T_C}\right)^{0.5}\right)\right]^2 \tag{5.2}$$

$$\alpha_0 = \frac{0.457247 R^2 T_C^2}{P_C} \tag{5.3}$$

$$n = 0.37464 + 1.54226\omega - 0.26992\omega^2 \tag{5.4}$$

$$b = \frac{0.07780 R T_C}{P_C} \tag{5.5}$$

Where T_C is the critical temperature, P_C is the critical pressure and ω is the acentric factor (Acentric factor is set as 0.2 for n-butane and 0.252 for n-pentane in Fluent).

The second change made to the simulation setup was the inlet temperatures. Before, the inlet temperature for the expander was at approximately ambient air temperature (300K), however, ORC systems work on the principle of

evaporating the fluid using low grade heat, so the inlet temperature will therefore be higher.

As ORC systems are closed systems and, as literature has alluded to, lubrication can be mixed into the refrigerant as an emulsion, the CFD simulations were performed for the Wankel expander with apex seals only. This is because in the previous simulations, the cases with apex seals performed much better, and if lubrication is less of a problem, seals should be used.

5.6.2. ORC CFD Results

The CFD results show very similar pressure trends to the previous simulations with air, as seen in Figure 5.27 (See Figure 5.13 in Section 5.3.4 for comparison). The near constant pressure inlet phase, the expansion phase and the low-pressure outlet phase can be seen clearly for all expansion chambers, showing the simulation is running as expected. The inlet mass flow rate shown in Figure 5.28 agrees with the pressure variations. It can be noted there are larger flow oscillations than air when the inlet port is first opened, which is logical when refrigerant is used due to its higher density and therefore higher fluid momentum. Finally, Figure 5.29 shows the specific entropy variation in one expansion chamber (corresponding to expansion chamber 2 in Figure 5.27). Here it is observed that there is very little variation, which is to be expected as the chambers are sealed during expansion and there is no heat transfer in the simulations. The spike in

specific entropy when the inlet port opens is due to the oscillations in inlet flow causing a spike in pressure above that of the inlet pressure. These three sets of results demonstrate the real gas model is working as expected.

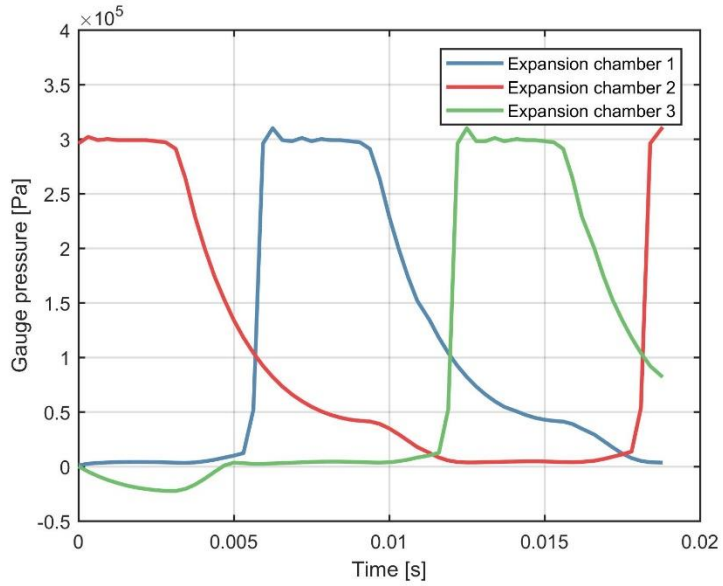


Figure 5.27 – Pressure variation in the expansion chambers in 1.5 revolutions (3bar, 4800rpm)

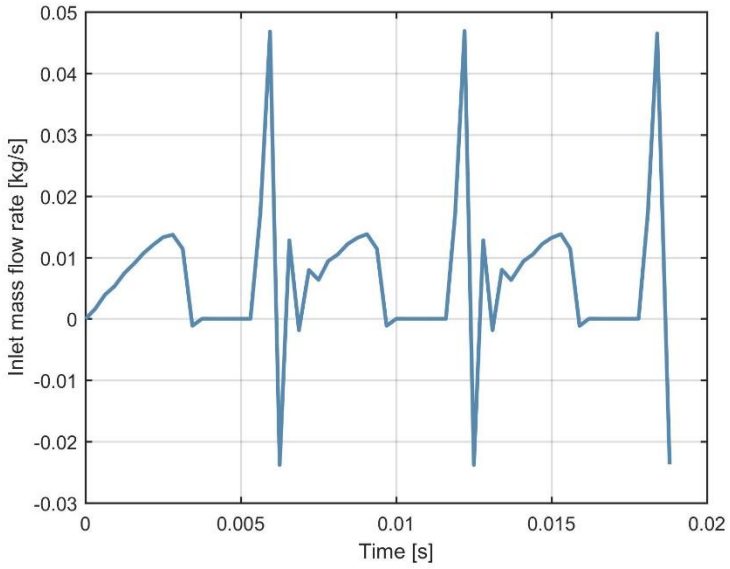


Figure 5.28 – Inlet mass flow rate during 1.5 revolutions (3bar, 4800rpm)

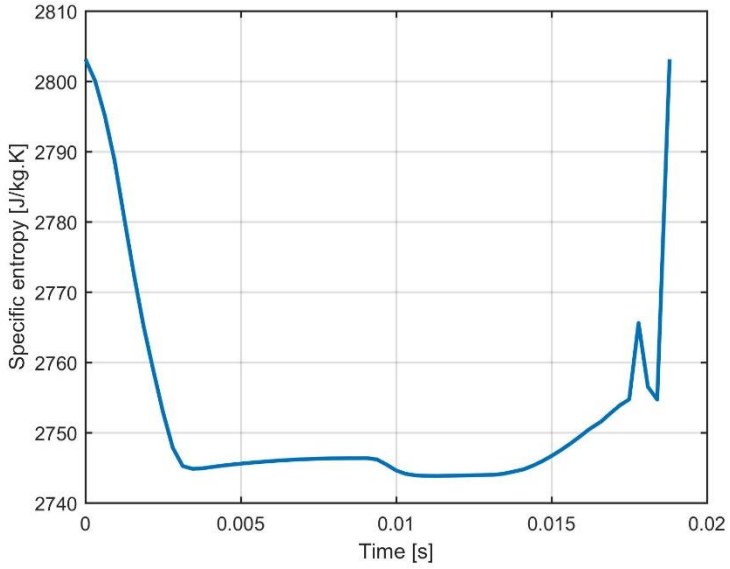


Figure 5.29 – An expansion chamber's entropy variation in 1.5 revolutions (3bar, 4800rpm)

To investigate how the expander performed using n-butane and n-pentane as working fluids, simulations were run with varying inlet gauge pressures (1-3bar) and speeds (3600-9600rpm). Figure 5.30 shows isentropic efficiency curves for the n-butane and n-pentane working fluids. For n-butane, 2bar inlet gauge pressure is the best in this case, showing the best performance at all speeds. However, the static shaft Wankel expander is easy to redesign to be optimum for an ORC system with any expander inlet pressure, by simply changing the angle the inlet is open for, which changes the expander's volume ratio. The maximum isentropic efficiency of 85% remains almost constant between 3600RPM and 6000RPM for the 2bar and 3bar cases. This suggests that there are negligible breathing problems in this speed range and efficiency losses due to the expander's breathing only become significant above 6000RPM.

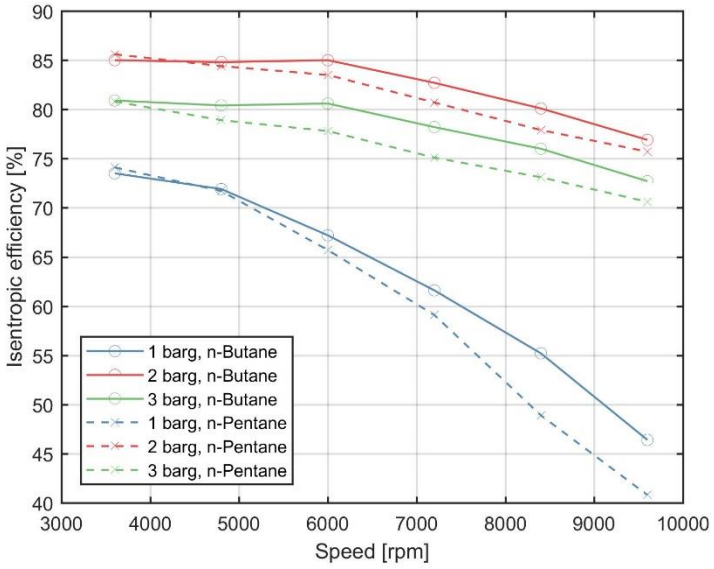


Figure 5.30: Isentropic efficiency against rotational speed for n-butane and n-pentane at different inlet pressures

For n-pentane, it is noticed that the trends are much the same as for n-butane, with the same maximum efficiency at the low speed cases. However, the efficiency curves decrease faster as the speed is increased. This is because of n-pentane's thermodynamic properties, which have a lower enthalpy drop for the same drop in pressure, therefore when it is producing a similar power output, a large inlet mass flow rate is required. Therefore, for this expander, under these operating conditions, n-butane would be the preferred choice of working fluid due to its efficiency. However, the choice in fluid should primarily be made considering the optimal operating temperatures of the attached ORC system.

Figure 5.31 shows that the 3bar inlet gauge pressure produces the highest power output, which is to be expected, as there is much higher available fluid power. The higher rotational speeds also resulted in larger power output,

showing again that breathing isn't much of an issue. Only the 1bar case showed a drop in the power output at the highest speeds, this is due to the pressure difference not being high enough to create the required inlet flow rate. The most efficient case at 85% (2bar inlet gauge pressure, at 6000RPM) gives 245W, which is significantly lower than the case with the maximum power output which gives 572W (3bar inlet gauge pressure, at 9600RPM). However, at the maximum power case the efficiency is significantly reduced to 73%. One option would be to select the 3bar inlet gauge pressure at 6000RPM, which serves as a middle ground with 80% isentropic efficiency and 413W power output.

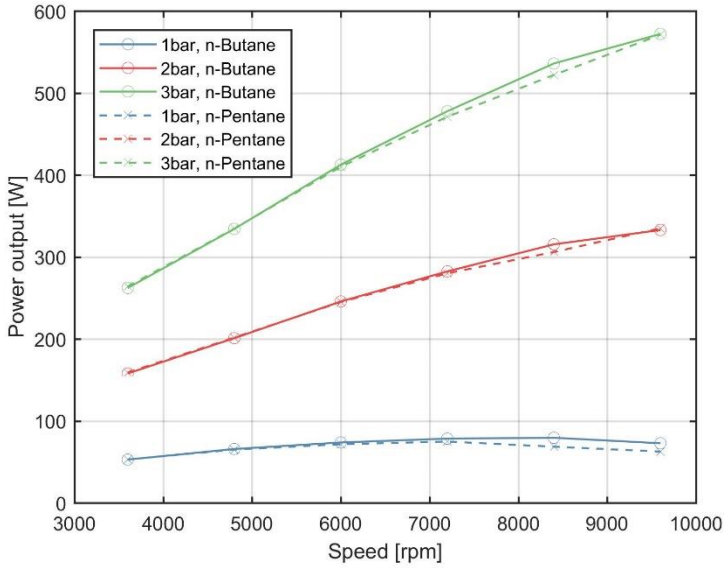


Figure 5.31: Power output against rotational speed for n-butane and n-pentane at different inlet pressures

Figure 5.33 shows the outlet temperature of the expander at different inlet pressures. It was found that the variation in expander speed did not create any significant difference in the outlet temperature. As can be seen, the larger

inlet pressures result in a slightly larger temperature drop, however the difference between 1bar and 3 bar is only 1.5K for n-butane and 2K for n-pentane. Therefore, the required outlet temperature does not need to be a consideration when selecting the inlet pressure, the inlet timing and the volume ratio both have a much larger effect on the outlet temperature. The temperature drop for n-butane (average of 21K) is larger than n-pentane (average of 17.12K), this is probably because n-butane has a lower boiling temperature for the same pressure and is therefore operating further from its two-phase region than n-pentane, this is shown in Figure 5.32.

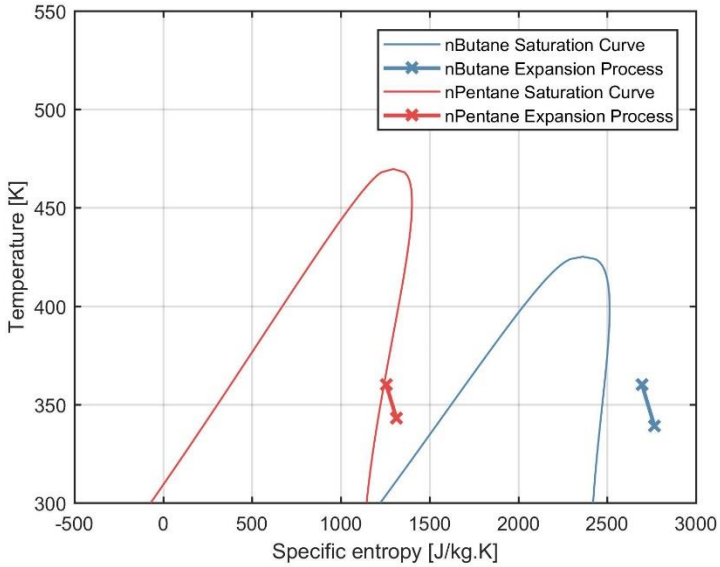


Figure 5.32 – TS diagrams with saturation curves for n-butane and n-pentane and points with the same inlet conditions

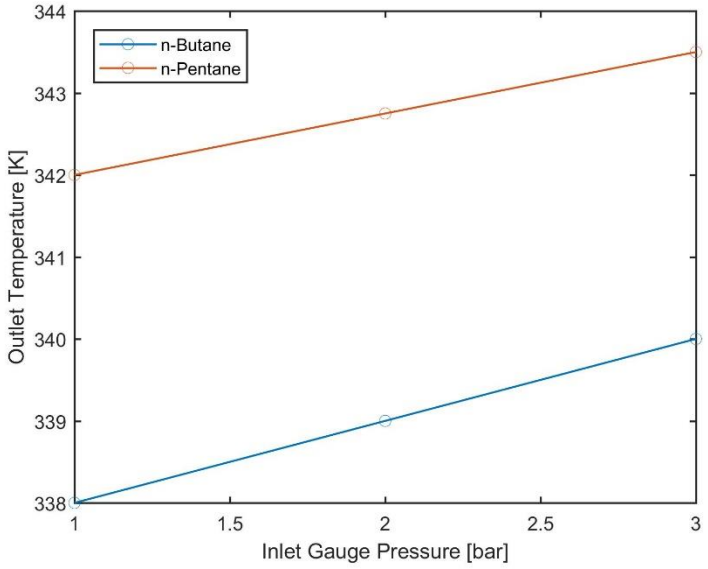


Figure 5.33 – Outlet temperature against inlet pressure for n-butane and n-pentane

Figure 5.34 shows how the pressure varies with the volume of a single chamber during one cycle. It compares the highest and lowest speeds, highlighting the increased pressure drop during the exhaust phase of the 9600rpm case. It is also seen that oscillations in the inlet phase cause a reduced pressure in the chamber during expansion. These factors lead to a reduced power output for the 9600rpm case, as the area within the pressure-volume diagram is reduced.

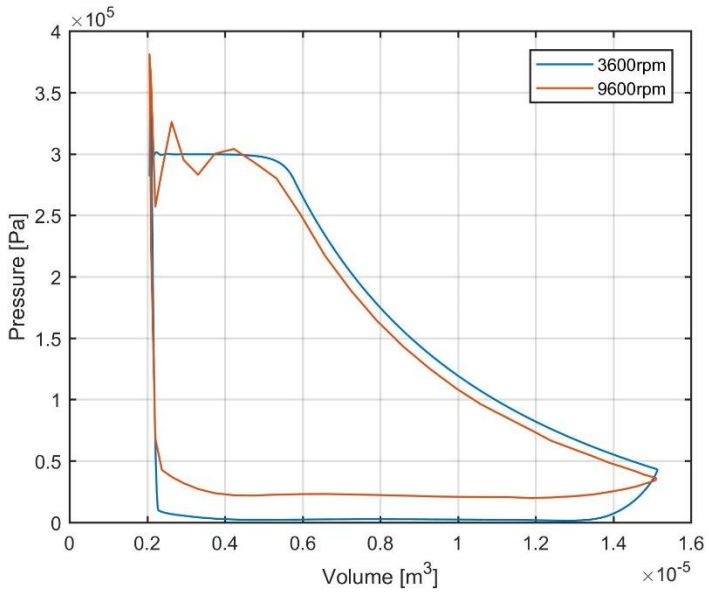


Figure 5.34 - Pressure-Volume diagram (n-butane, 3bar)

Figure 5.35 shows similar pressure-volume diagrams, this time comparing n-butane and n-pentane under the same conditions. Both working fluids exhibit very similar shapes, however n-pentane appears to be shifted upwards, presenting a larger pressure drop during the outlet phase. N-Pentane also has a different oscillation pattern during the inlet phase, due to its increased density and viscosity causing different rates of fluid acceleration. In this case it causes a higher pressure during the expansion phase, which ultimately causes the two working fluids to have very similar power outputs.

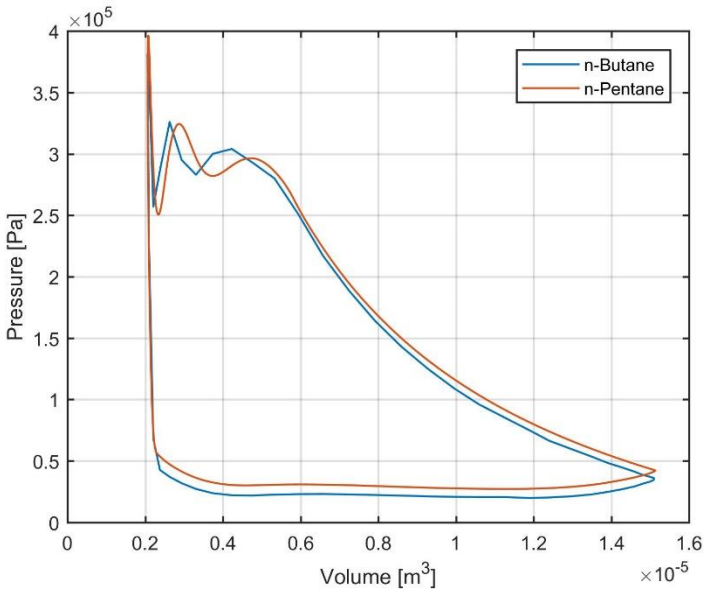


Figure 5.35 – Pressure-Volume diagram (9600rpm, 3bar)

Figures 5.36 and 5.37 show velocity contours within the expander's chambers, both at the same time and operating conditions, for the two different working fluids. N-Butane shows the highest inlet velocity at this time step, however when looking at Figure 5.38 it is noticed that the average inlet velocity over the inlet period is very similar for both working fluids. This suggests that the losses due to viscous effects do not have much effect in the inlet port, which agrees with the results in the pressure-volume diagrams above.

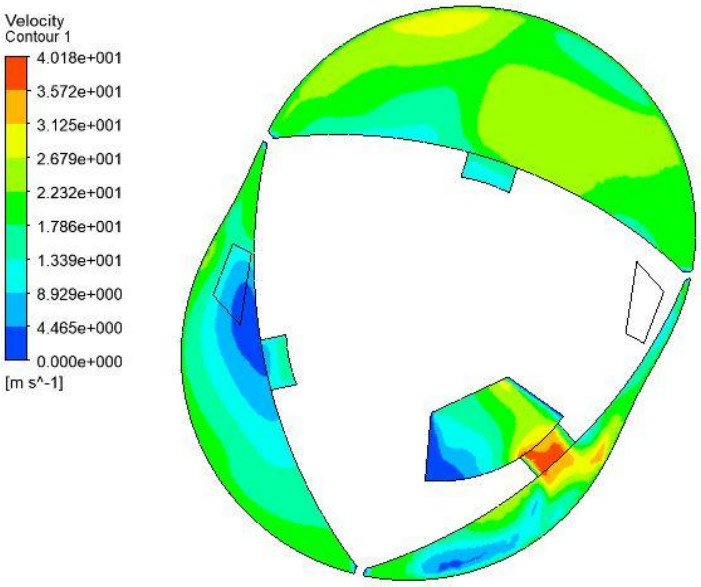


Figure 5.36 – Velocity contours during a chamber's inlet phase (n-butane, 9600rpm, 3bar)

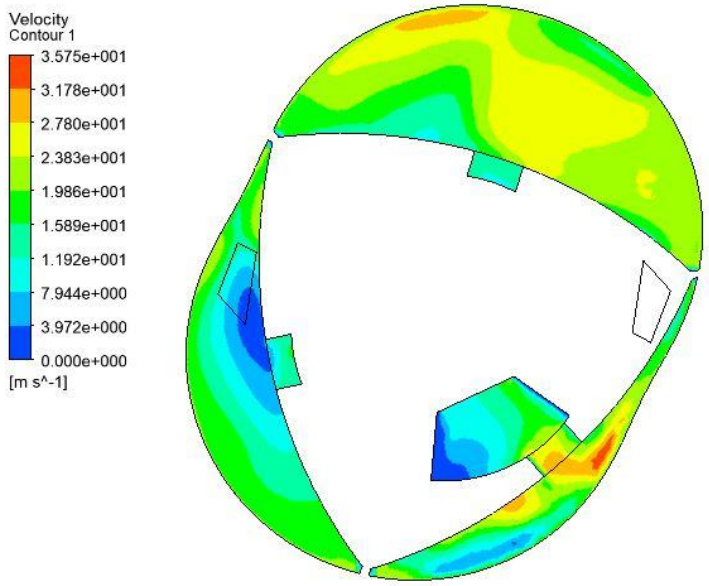


Figure 5.37 – Velocity contours during a chamber's inlet phase (n-pentane, 9600rpm, 3bar)

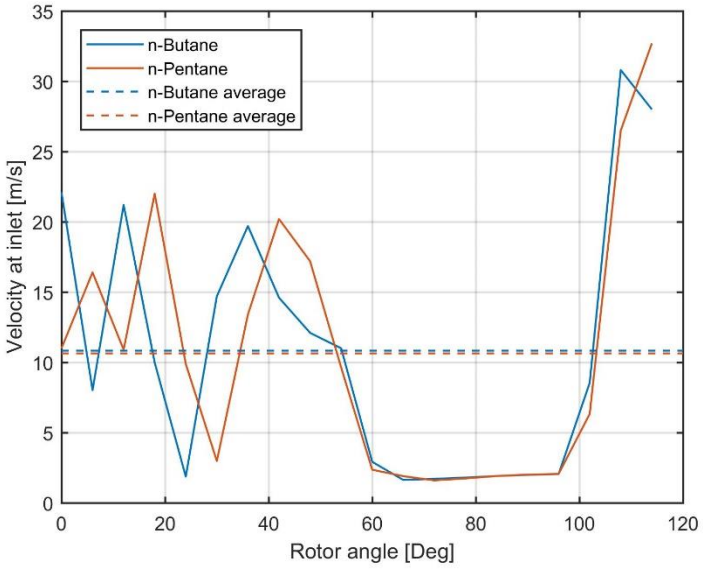


Figure 5.38 – Fluid velocity at the inlet against rotor angle

Figures 5.39 and 5.40 show Temperature against entropy diagrams for the expansion process with n-butane and n-pentane. Both working fluids are 'dry', meaning that during expansion the fluid cannot enter the two-phase region. To best match the ideal organic Rankine cycle, the expansion process should begin close to the saturation curve, this allows the preceding evaporation process to occur at a more constant lower temperature. As the organic Rankine cycle is preferred for the use with low grade heat, this is an important factor. Therefore, n-pentane may be the more desirable choice of working fluids if an organic Rankine cycle were to be operated at the operating conditions used in this study.

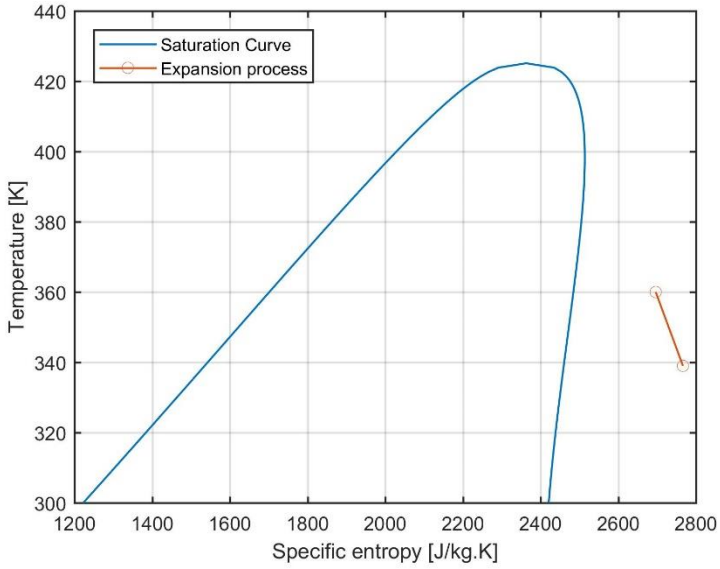


Figure 5.39 – Temperature -Entropy diagram showing expansion process for n-butane

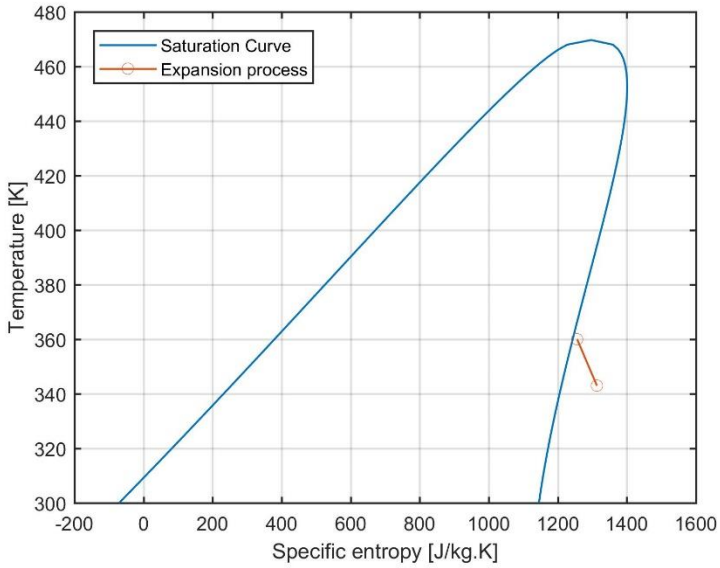


Figure 5.40 - Temperature -Entropy diagram showing expansion process for n-pentane

5.6.3. Practical Considerations

Some practical considerations are needed if the expander is to be utilised in an ORC system rather than a liquefaction system. Firstly, as leakage is a much more important factor when using organic fluids, the seals would need to have much more analysis, to ensure no leakage occurs. This could lead to

larger, tighter fitting seals or multiple seals in place of one both of which could cause an increase in frictional losses.

A further consideration would be the choice of some suitable lubricant for the seals, bearings and gears that would be mixed with the organic fluid. The choice of this lubricant would be fairly complicated, as it would have to take into account the materials used, the bearings, how well the lubricant mixes with the refrigerant, the lubricant's effect on other components in the ORC system (pump, heat exchangers) and how the lubricant would cope with the different temperature and pressure conditions in the ORC.

Finally, the materials of the expander would have to be analysed and possibly changed if the temperature would cause any undesirable effects. For example, this could be differences in thermal expansion causing increased friction.

5.7. Chapter Summary

- The results are given from CFD parametric studies performed on both the static shaft Wankel expander and the standard Wankel expander for comparison.
- The standard Wankel expander was shown to have inadequate isentropic efficiency even at optimal operating conditions and with the improved double-sided ports design (64.88% isentropic efficiency, 100W power output).

- The static shaft Wankel expander significantly improved the maximum isentropic efficiency to 87.35% with 693W power output for and expander with the same dimensions.
- Efficiency at different inlet pressures and inlet timings are linked to each other for the static shaft Wankel expander: it can be designed for a large range of inlet pressures.
- Isentropic efficiency increases with speed for all cases without apex seals because of the reduced time for leakage.
- Isentropic efficiency decreases slightly with speed for cases with apex seals because of the pressure drop across the inlet and outlet ports.
- Reducing apex clearance leads to increased isentropic efficiency with an exponential relationship, achieving small apex clearances is therefore vital if apex seals are not used.
- The two modifications were made to the CFD model, the gas model was changed from ideal gas to a real gas model and the inlet temperature of the expander was increased to better match the conditions in an ORC system.
- The maximum isentropic efficiency achieved was similar (85% isentropic efficiency) to the simulations with air as the working fluid, although at a lower power (245W power output). This implies little modification would be required to use the Wankel expander for different applications.

- The results using n-butane and n-pentane had very similar isentropic efficiencies, but n-pentane may be the better choice if used in an ORC system with these operating conditions as the expansion process occurs closer to its saturation curve.

CHAPTER 6

EXPERIMENTS: TEST RIG

6.1. Chapter Introduction

This chapter describes the design and setup of an experimental rig to test the performance of two static shaft Wankel expander prototypes, one manufactured from plastic and the other from metal. The layout of the rig is detailed and the components are fully described. The calibration processes of the various sensors are described, and their uncertainties are calculated. Finally, the test procedures are given.

6.2. Experimental Rational

The design and experimental analysis of a Wankel expander prototype was decided on primarily for two reasons. The first reason was to provide validation of the CFD simulations. Although the convergence of the CFD solutions were ensured and the CFD mesh was sized to reduce its contribution to errors, there is no true way to check the CFD model accuracy without having actual experimental results to compare. The literature provides a very incomplete picture of the Wankel expander's experimental performance, with most omitting crucial data/results that would be needed for validation such as efficiency or the expander dimensions. Therefore, the most reliable way forward was to design and carry out experiments.

The second reason was to provide proof of concept for both the new design without apex seals and the new design utilising the static shaft concept. There has been no known published literature where a Wankel expander without apex seals has been tested experimentally, meaning there is no sure way of knowing if some completely overlooked problems will prevent it from working, without testing experimentally. Similarly, there is no known published literature on any static shaft Wankel device, so for the same reasons as above, experimental testing is desirable.

6.3. Experimental Test Rig

An experimental test rig was designed and setup in order to evaluate the developed static shaft Wankel expanders' performances. The test rig consisted of the following:

- Static shaft Wankel expander
- Compressed air inlet and outlet pipes
- A pressure regulator and air filter
- A gas flow meter
- Inlet and outlet pressure transducers
- Inlet and outlet thermocouples
- A Datum torque meter
- An output shaft brake and load cell
- An Arduino Mega 2560 board
- A PicoLog TC-08 thermocouple data logger

- A PicoLog 1012 data logger with a small terminal board
- A computer running windows 7 with PicoLog 6 and MATLAB software installed
- A laboratory power supply set to 8V

Figure 6.1 shows a schematic diagram of the test rig setup and the connections between the different components. Figure 6.2 shows a photo of the components situated on the test bench.

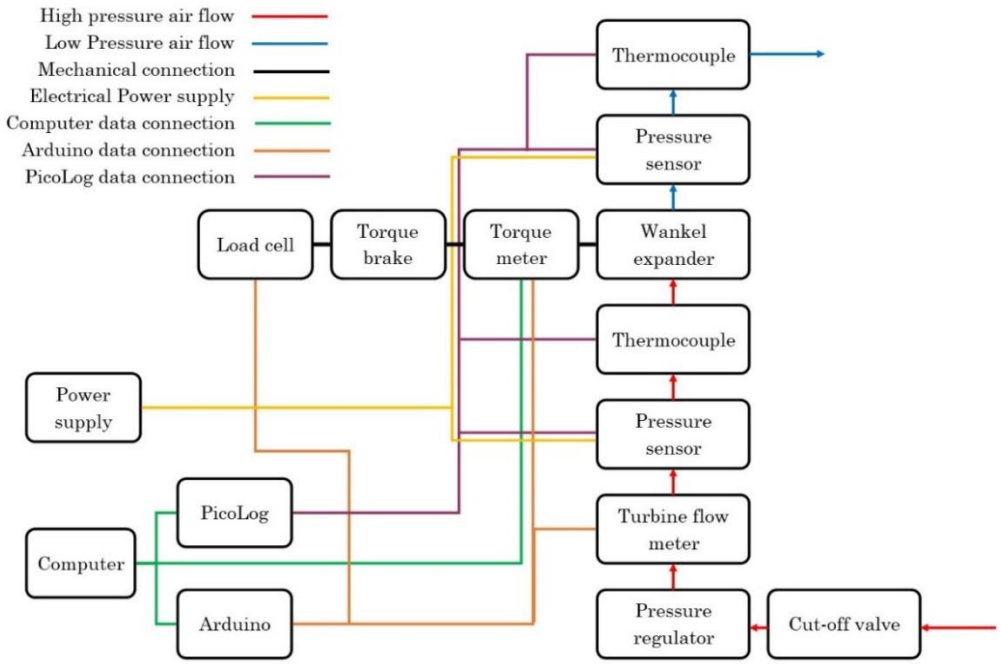


Figure 6.1 – Schematic diagram of the experimental test rig setup

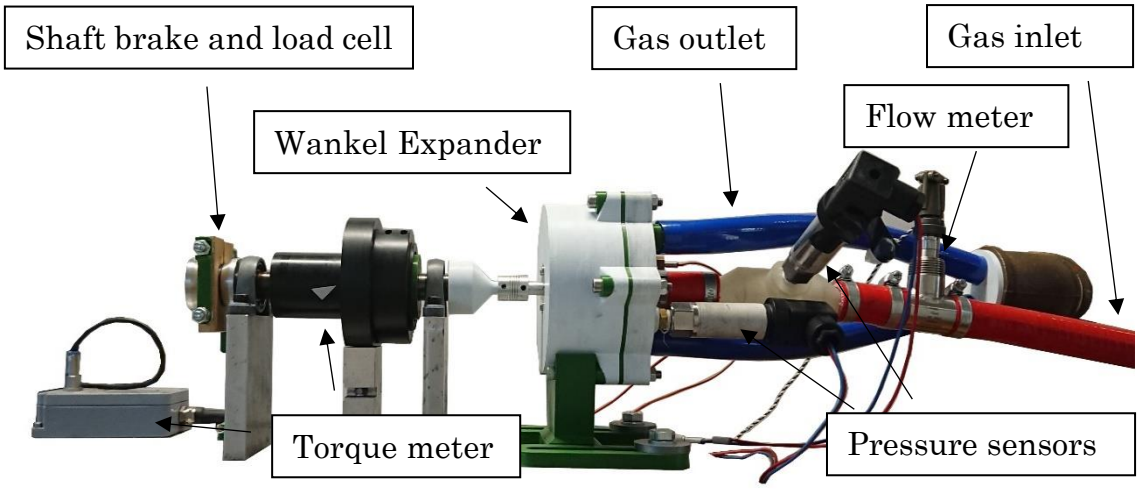


Figure 6.2 – The Wankel expander experimental test rig

6.3.1. Compressed Air Input

All experimental testing was carried out using compressed air as the working fluid. A compressed air line was routed to the experimental rig from an HPC SM11 Rotary screw air compressor (rated max pressure of 8bar and max flow rate of 40.4CFM) which supplied compressed air to the laboratory building. At the connection point to the expander, the compressed air line could supply a maximum of 2bar static gauge pressure with no air flow, however the available pressure is reduced to approximately 1.2bar static gauge pressure with air flow, mainly due to flow losses in the long piping between compressor and test rig location. The compressed air temperature tends to be similar to the ambient air temperature, meaning there is some deviation day to day.

Prior to the compressed air line's connection to the rig, a pressure regulator is attached in-line. The pressure regulator allows the pressure to be set to a value which then remains relatively stable, thus allowing different tests with the same inlet pressure to be carried out. The pressure regulator also has a

built-in filter and water trap which ensure clean dry air is supplied to the Wankel expander, removing a possible source of error in results and possible failure in operation. There are also two further in-line valves before the regulator that provide a backup method of shutting off the flow, in case of failure in the regulator.

After the pressure regulator, a flow meter and a manifold for the inlet pressure and temperature sensors are attached in-line. Connecting the sensors manifold and the expander is a small section of flexible hose, rated up to 8bar pressure. The flexible hose allows for slight misalignment of the fixed expander and the fixed piping making assembly and disassembly much quicker and easier. Furthermore, the flexible piping helps isolate the sensors from any vibrations from the Wankel expander during operation.

6.3.2. Temperature Measurement

Measurement of the inlet and outlet temperatures was achieved through use of T-type thermocouples with 0.75mm diameter probes, which are placed in the air flow. Each had a ¼" male BSPT fitting so they could be inserted into the flow without causing leakage. The thermocouples were connected to a PicoLog TC-08 datalogger, which was in turn connected to the Windows 7 computer where the data was recorded for each test using PicoLog 6 software. The PicoLog TC-08 also provided an inbuilt thermometer which provided the reference point for the thermocouples and allowed the measurement of the ambient temperature in the laboratory.

The accuracy of the T-type thermocouple is frequently quoted as ±1°C [110], this was verified for the thermocouples using a water bath with a thermostat heater. The thermocouples and a spirit thermometer were placed in the water and the deviations between the readings were recorded as the water temperature was raised in increments. Figure 6.3 shows the results of these tests with R^2 values of 0.9978 for the inlet thermocouple and 0.9983 for the outlet thermocouple.

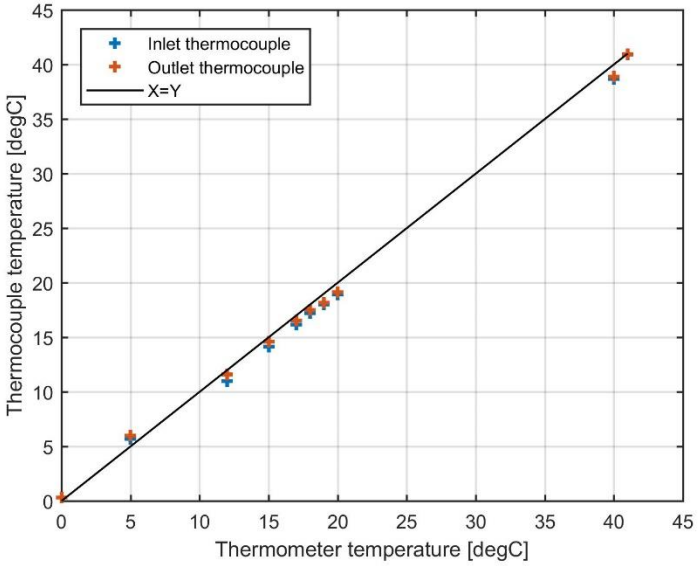


Figure 6.3 –Temperature verification of thermocouples

6.3.3. Pressure Measurement

Air pressure was measured at the inlet and the outlet of the Wankel expander, at the same locations as the temperature measurements. Two UNIK 5000 series absolute pressure transducers with pressure ranges of 0-10bar were used. The pressure transducers were powered by a variable power supply set to 8V and had outputs of 4-20mA which were read using a Pico PP545 Small Terminal Board connected directly to a PicoLog 1012 voltage

data logger. In order to read current instead of voltage, the Pico Small Terminal Board was modified from its default setup. This involved soldering a 'shunt resistor' to the board which allowed voltage at the data logger input to vary instead of the current. This resistor had to be chosen specifically in order to ensure the voltage range into the data logger was optimum. The data logger had a maximum voltage input of 2.5V to prevent damage, however if the voltage range was too small, the readings would be much less accurate. Therefore, the resistor had to be chosen to allow a maximum voltage of 2.5V. Equation 6.1 was used to find the resistance value.

$$R = \frac{2.5V}{I_{max}} \tag{6.1}$$

Where R is the resistance, V is the transducer supply voltage and I_{max} is the maximum current the transducer will produce. Using this equation, the optimum resistance was found to be 1kΩ.

In order to convert the voltage reading to pressure, the pressure transducers were calibrated using a water-based pressure calibrator. This device had a pipe connection between the pressure transducer, a pre-calibrated pressure gauge, and a cylinder/plunger arrangement that could be used to apply pressure. A screw was turned to adjust the plunger and apply different pressures to the water trapped within. This was done for multiple calibration points where the pressures from the gauge and the corresponding voltages were recorded. Finally, the pressure-voltage values were plotted in excel, where a linear trend lines were calculated both with R^2 values of 0.9999,

which indicates a high accuracy of the trend lines. Figure 6.4 shows the calibration data and trend lines for both transducers.

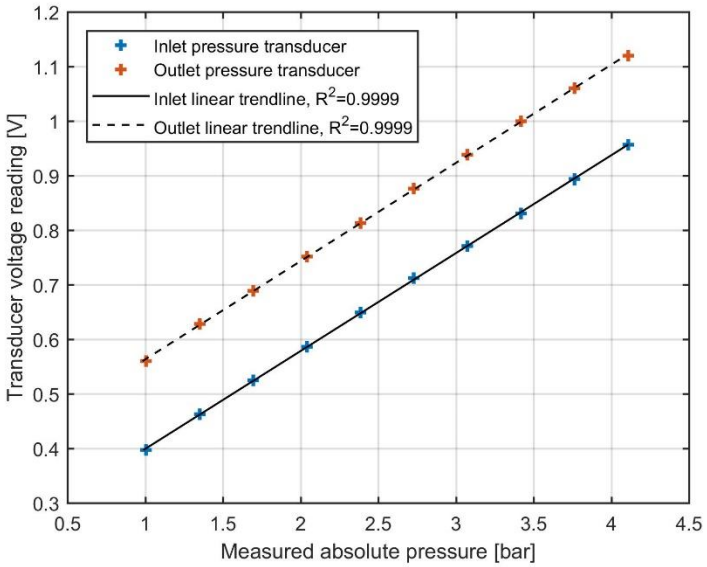


Figure 6.4 – Pressure transducer calibration data

The voltage measurements from the datalogger were read by the same software as the temperature data logger (PicoLog 6). In this software the output voltages of the pressure transducers were altered using the trendline equations to show the accurate pressure readings.

6.3.4. Flow Rate Measurement

Several different methods for flow rate measurement were utilised during experiments. Firstly, a Key Instruments variable area flow meter for air, with a measurement range of 400-4,000LPM was fitted. However, this flow meter was only accurate to the nearest 200l/min. As the experiments generally only spanned a flow range of 400l/min this accuracy level was not sufficient.

The second method of flow measurement utilised was a pitot-static tube combined with a differential pressure sensor. This allowed the static and total pressures of the inlet flow to be measured and from this the velocity of the fluid calculated. Unfortunately, the results from this device during testing gave an unexpected trend, not matching with the CFD results or the readings from the variable area flow meter. This deviation occurred at certain operating conditions and was thought to be due to either local flow patterns creating low pressure areas in the pipe or the inlet flow oscillations causing inaccuracies at certain frequencies.

Therefore, an Omega FTB-933 gas turbine flow meter (with a measurement range of 1-10ACFM) was found and incorporated into the experimental rig. This flow meter's output was the frequency of an oscillating voltage and the meter was supplied with a calibration sheet showing accuracy to be within ±1%. Using this flow meter, the flow rate readings followed both the trend of the CFD and the variable area flow meter, adding further confirmation that the readings were accurate. However, as this flow meter was found and not specified for this experiment's specific operating limits, its maximum flow rate was too low for most of the desired operating conditions.

To solve this problem, a larger gas turbine flow meter was purchased, which was specified for a flow rate high enough for all planned operating conditions. This flow meter was a BGFT-25 Gas Flow Turbine Meter form Bell Flow Systems with a measurement range of 2.5-25m^3/hour. The same electrical output was produced in the form of a millivolt sine wave, with a frequency in

proportion to the flow rate. A quoted ±1.5% accuracy was given by the calibration certificate. To read the sensor's signal, an Arduino Mega 2560 combined with a Windows 7 computer was used. This was done by writing an Arduino script to count the number of times the signal went from being zero to a positive number in a second (which found the frequency) and then converting it to flow rate using the equation derived from the 'k factors' from the flow meter's calibration sheet. Figure 6.5 shows the calibration data points and derived linear trendline.

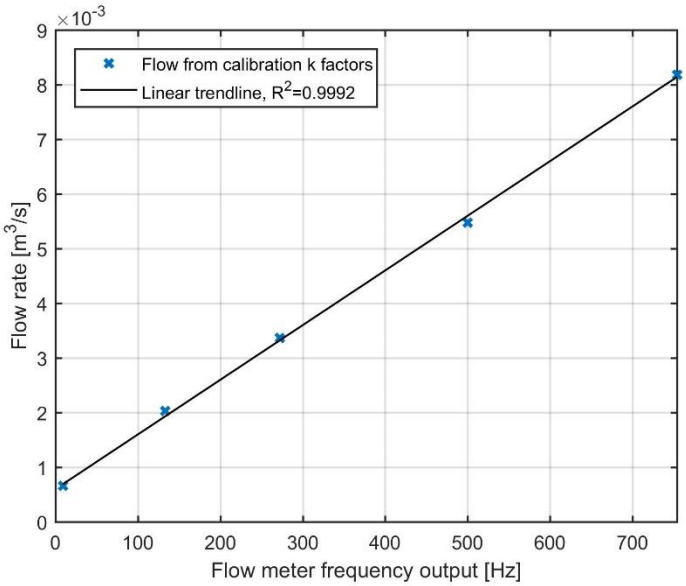


Figure 6.5 – Flow meter linear relationship derived from calibration 'k factors'

To record the output of the Arduino to the computer, a MATLAB script was written. The Arduino is connected to the computer via USB, which the Arduino transmits a serial data connection through. The MATLAB script can then use functions to read the data from the serial connection. The serial data comes in lines of strings, these were split into separate data (as torque and speed were also read by the Arduino) and then converted into number format.

After data recording is finished the MATLAB script output the recorded values into a CSV file with an appropriate name.

6.3.5. Torque Meter

A Datum torque meter, shown in Figure 6.6, was fitted between the output shaft of the expander and the brake. The torque meter gave outputs of torque, speed, and power. The torque meter could be directly connected to a computer, where Datum's proprietary software can read and record data, or it could give outputs in the form of a 0-5V voltage signal.

Figure 6.6 – Datum torque meter with signal conditioner

The accuracy of the torque meter was tested using manual measurements of the torque. Two forms of manual measurement were carried out, both utilising spring scales. The first form was static measurement, where a spring scale was used to apply a known force to a known radius arm connected to the shaft of the torque meter, in these tests the torque meter gave an average

error of 52%. The second form was a dynamic measurement, where two spring scales were used to pull a belt against a pulley on the output shaft, as demonstrated by Figure 6.7. The difference between the two spring scale readings gave the friction on the pulley radius, which was used to calculate the torque being applied. In these tests the torque meter gave an average error of 79% over a variety of speeds. These errors were believed to be caused by internal damage in the device. Due to the large inaccuracies of the torque meter readings, it was decided not to use the torque output.

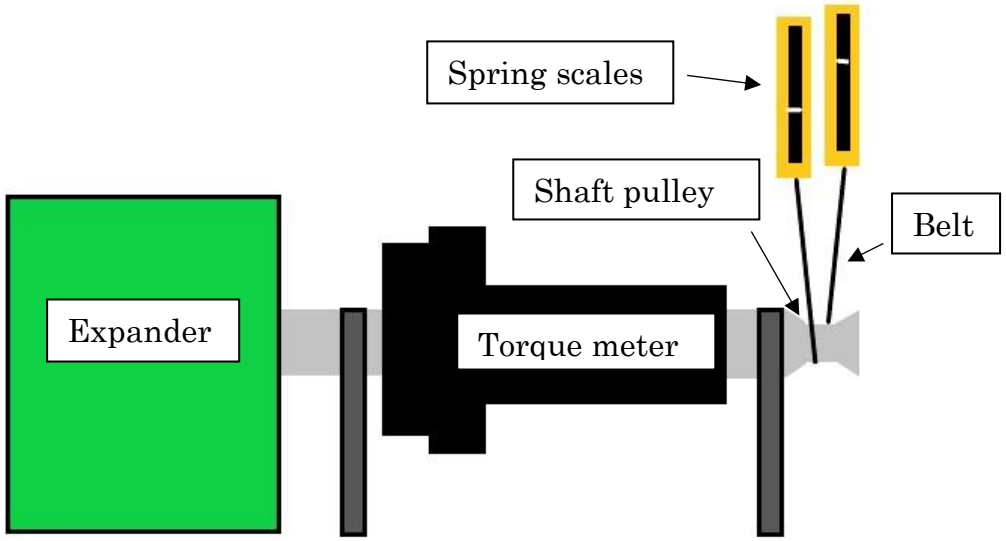


Figure 6.7 – Diagram showing the testing of torque meter with two spring scales

The torque meter device was still utilised for the rotational speed reading. The speed measurement was accurate up to 20,000RPM according to the supplier and no inaccuracies were found during checks. The rotational speed output from the torque meter was still utilised via the voltage output signal. This voltage was read by the Arduino, which used a calculated factor to convert the voltage into RPM, the factor was calculated using Figure 6.8. The

Arduino was decided on over the proprietary Datum software so the data could be recorded in the same place as the load cell and flow meter measurements.

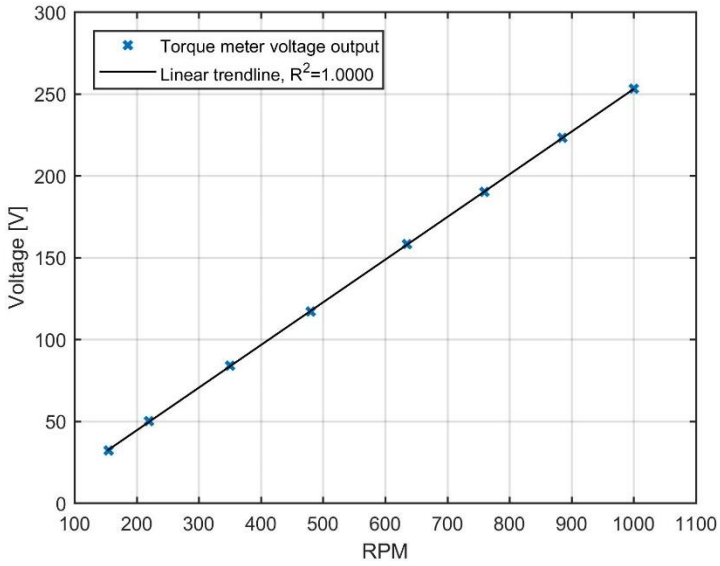


Figure 6.8 -Calibrating the analogue voltage output for the rpm

6.3.6. Load Cell and Brake

To control the torque applied to the expander's output shaft, a 'Prony' brake setup was utilised. This allowed the speed of the expander to be controlled over a wide range of speeds and torques. The torque arm of the Prony brake pushed against a load cell which was attached to the test bench. This allowed the measurement of the force it exerted and, as the perpendicular distance of the torque arm was known, the exerted torque could be calculated.

The Prony brake consisted of two wooden brake blocks that had curved surfaces and each with two bolt holes drilled through. The curved surfaces were clamped against an aluminium pulley connected to the output shaft with

two bolts. One bolt was simply secured with two nuts locked together, the second had one nut that was captured within a 3D printed part which stopped it rotating. This allowed the torque resistance to be adjusted easily during operation by screwing that bolt in or out, which adjusted the wooden brake pad clamping force. To increase the stability of the torque resistance, a small spring was placed between the captured nut and its corresponding brake pad, this allowed the torque to be applied more smoothly when the bolt was tightened. Figure 6.9 shows the Prony brake with load cell setup.

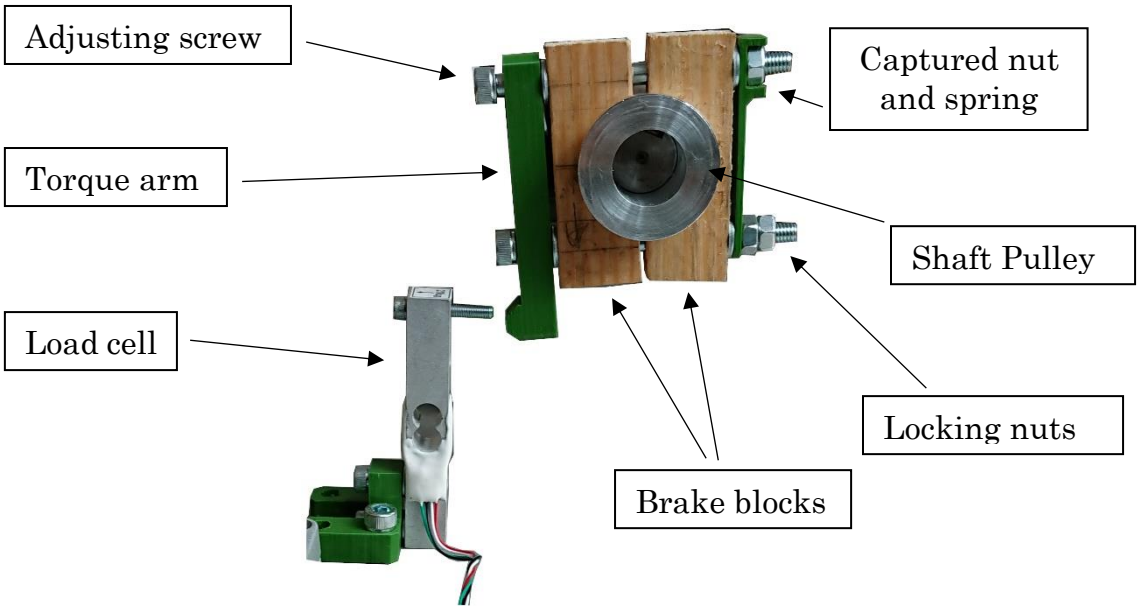


Figure 6.9 – Load cell and brake

The load cell was of generic type and could handle a maximum force of 50N (translating to a maximum torque of 2.35Nm). It was connected to an HX711 load cell amplifier circuit, which was in turn attached to the Arduino. The Arduino utilised a modified version of code released in an Arduino library [111] to read the HX711 output signal. To calibrate the load cell, different forces were applied in the normal direction on the load cell using a spring

scale. The Arduino output from the load cell was then recorded for each calibration point, Figure 6.10 shows the calibration.

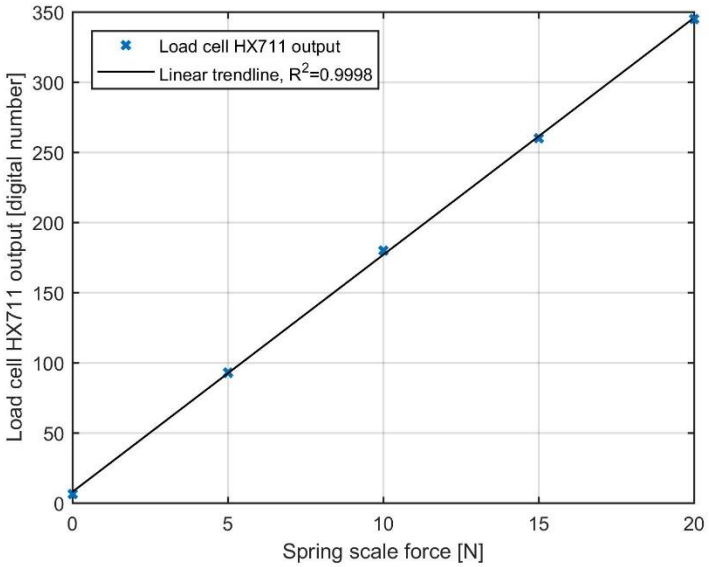


Figure 6.10 – Load cell calibration with spring scales

6.3.7. Calculation of Power Output and Isentropic Efficiency

The power output and the isentropic efficiency are the two main performance parameters that are used to analyse an expander. The Equations 6.2 to 6.6 describe how the power output and isentropic efficiency of the expander were calculated from the sensors of the test rig.

$$P = 2\pi\omega\tau \tag{6.2}$$

Where P is the brake power output, τ is the torque output from the load cell on the brake and ω is the rotational speed in rev/s from the torque meter.

$$\dot{m}_{inlet} = \frac{\dot{V}_{inlet} p_{inlet}}{R T_{inlet}} \tag{6.3}$$

Where $\dot{m}_{inlet}$ is the inlet mass flow rate, $\dot{V}_{inlet}$ is the inlet volumetric flow rate, p_{inlet} is the inlet absolute pressure, T_{inlet} is the inlet temperature and R is the specific gas constant for air.

$$w_{actual} = \frac{P}{\dot{m}_{inlet}} \tag{6.4}$$

Where w_{actual} is the measured specific work output.

$$w_{ideal} = h_{inlet} - h_{outlet,isentropic} \tag{6.5}$$

Where w_{ideal} is the ideal specific work output, h_{inlet} is the inlet specific enthalpy and $h_{outlet,isentropic}$ is the specific enthalpy calculated from the outlet pressure and the inlet entropy using the CoolProp library.

$$\eta = \frac{w_{actual}}{w_{ideal}} \tag{6.6}$$

6.3.8. Sensor Uncertainty and Uncertainty Propagation

The uncertainties for all the sensors were calculated and then used to find the total uncertainty propagation on the calculated performance parameters of power output and isentropic efficiency. For each sensor the uncertainties due to random and systematic errors were found and used to calculate the total uncertainty as shown in Equation 6.7. The uncertainties due to random errors were calculated for each sensor using the mean deviation of calibration data from the trendline multiplied by Student's distribution factor (95th percentile confidence interval) for the number of sample points. Equations 6.8 and 6.9 show the calculation of sensor uncertainties due to random errors.

Uncertainties due to systematic errors were found from either the accuracy of the associated calibration gauges accuracies or the accuracy quoted on the calibration certificates, if available.

$$U_{total} = \sqrt{U^2{}_{random} + U^2{}_{systematic}} \tag{6.7}$$

Where U_{total} is the total sensor uncertainty, U_{random} is the uncertainty due to random errors and $U_{systematic}$ is the uncertainty due to systematic errors.

$$U_{random} = t_{(N-1),95\%} S_{\bar{x}} \tag{6.8}$$

Where $t_{(N-1),95\%}$ is the 95th percentile confidence interval Student's distribution factor and $S_{\bar{x}}$ is the mean deviation.

$$S_{\bar{x}} = \frac{1}{\sqrt{N}} \sqrt{\frac{\sum_{i=1}^{N} (x_i - \bar{x}_i)^2}{N-1}} \tag{6.9}$$

Where N is the number of calibration sample points, x_i is a calibration sample point at i and $\bar{x}_i$ is the value on the fitted trendline at i. The calculated sensor error values are presented in Table 6.1.

Table 6.1 – Random, systematic and total errors for sensors

Sensor	**Random error, U_{random}**	**Systematic error, $U_{systematic}$**	**Total error, U_{total}**
Inlet temperature	0.6549°C	1.0000°C	1.1954°C
Inlet pressure	685.5Pa	1725.0Pa	1856.2Pa
Outlet pressure	817.5Pa	1725Pa	1908.9Pa
Inlet flow rate	0.0001051m^3/s	0.0001250m^3/s	0.0001633m^3/s
Outlet shaft rotational speed	0.1377rad/s	0.1047rad/s	0.1730rad/s
Outlet shaft torque	0.006747Nm	0.04700Nm	0.04748Nm

Error propagation was then calculated using Equations 6.10 and 6.11. Microsoft Excel was used to find the total propagation through all the functions required to calculate the power output and efficiency values. This resulted in a power output uncertainty of ±2.79W and isentropic efficiency uncertainty of ±0.523%.

$$y = f(x_1, x_2, \dots, x_n) \tag{6.10}$$

Where y is the output to any function f with any number of input variables $x_1, x_2, \dots, x_n$.

$$U_y = \sqrt{\sum_{i=1}^{n} \left(\frac{\partial y}{\partial x_i} U_{x_i}\right)^2} \tag{6.11}$$

Where U_y is the uncertainty of output y, $\frac{\partial y}{\partial x_i}$ is the partial differential of y with respect to input variable x_i and U_{x_i} is the uncertainty of input variable x_i.

6.3.9. Safety

Due to the nature of the experiments, proximity to high pressure gases and high-speed rotating machinery was inevitable. To ensure the safety of the user, various steps were taken to reduce the risk of injury. The first was the use of safety goggles whenever a test was running. Aside from this everyone within the lab wore lab coats to cover any possible loose clothing, to prevent it getting caught. To protect from the possibility of an explosion of compressed air or a rotating part being flung, a guard was used to cover the expander and shaft. A long Allen key passed through a small hole in the guard allowed the torque brake to be adjusted safely during operation. Finally, the outlet from the expander was attached to a large pipe that was routed outside, to reduce noise levels in the lab. Ear defenders were also available if the noise became too loud, at the user's discretion.

6.3.10. 3D Printed Parts

Some of the parts used in the experimental rig were 3D printed to keep costs and setup time low. The first of these parts was the outer casing for the expander, shown in Figure 6.11. This part was not subject to any significant air pressure or other forces, so the plastic 3D printed material was sufficient. This part held the expander in place with two places to press fit bearings on the interior and two feet that were bolted to the test bench on the exterior. It also provided two protruding pipe sections to act as the outlet flow manifold

and two further BSPT threaded holes into which the outlet pressure transducer and thermocouple were located.

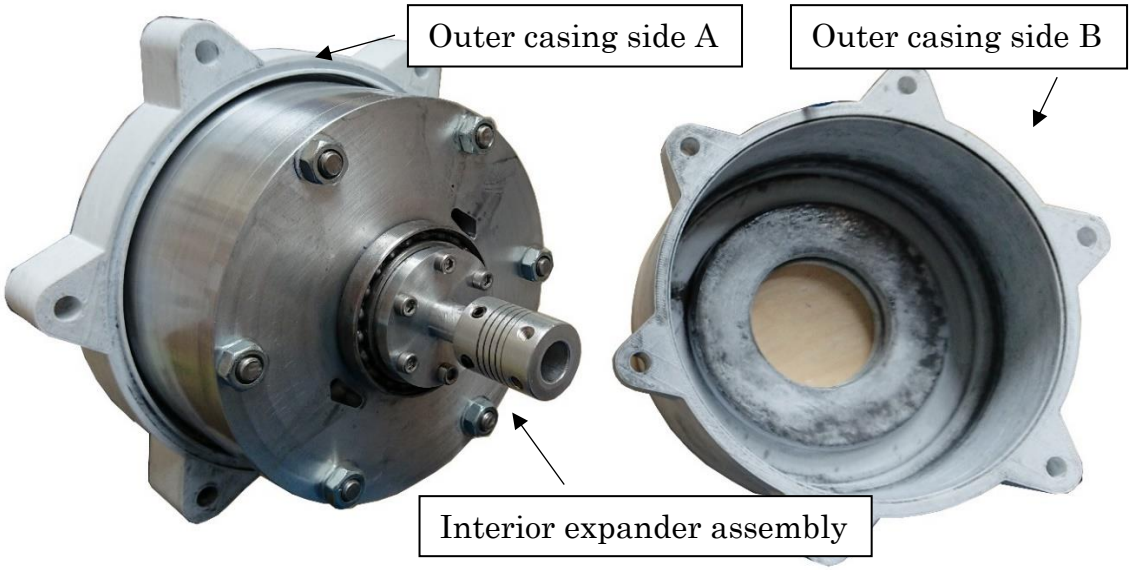
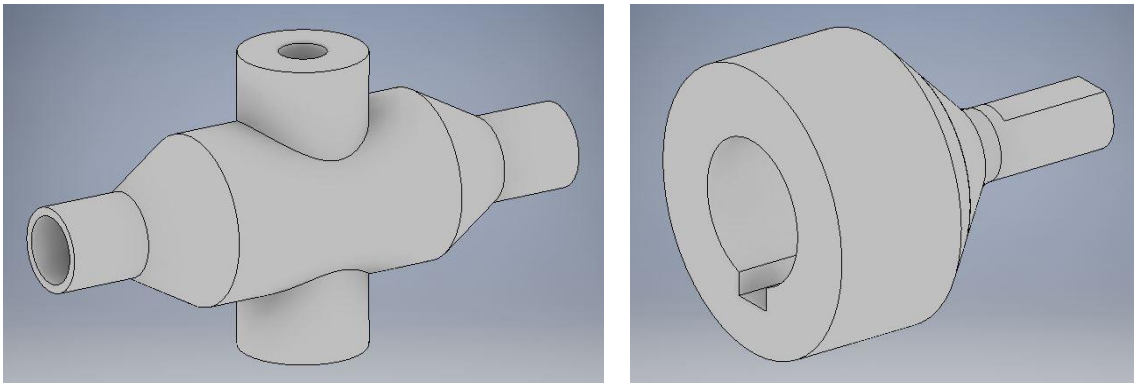


Figure 6.11 - Expander inside the 3D printed plastic casing halves

The next 3D printed part is the inlet sensors' manifold which allowed connection of two sensors to the inlet flow with ¼" BSPT fittings. This part was successfully leak tested at the maximum pressure the compressed air line could achieve (Approx. 2bar). The manifold is shown as a CAD image in Figure 6.12a. At first this allowed for the connection of a pitot-static tube, which is why it has an enlarged section in the middle, however, later it simply contained a static pressure transducer and a thermocouple.



(a) (b)

Figure 6.12 – 3D printed parts (a) inlet sensors manifold (b) Expander to torque meter coupling

The coupling between the expander outlet and the torque meter was also a 3D printed part, shown as a CAD image in Figure 6.12b. Although this part does experience relatively high stresses, it was considered a good contingency to have a part in the coupling that should fail long before anything else, thus protecting some of the more expensive parts.

The final few printed parts included the torque arm of the brake, the part to capture the nut (as previously mentioned) and a mount to connect the load cell to the test bench. None of these parts experienced high stresses and therefore performed perfectly well.

6.3.11. Experimental Procedure

1. Turn on computer and power supply
2. Open PicoLog6 software and load calibration settings
3. Open MATLAB and load Arduino recording code
4. Set the expander's brake to fully off (to help starting)
5. Drain any liquid from air regulator, turn regulator to off position
6. Open both shutoff valves

7. Slowly open pressure regulator until the expander starts rotating
8. Check all sensors are working
9. If low desired inlet pressure: Open pressure regulator further until desired inlet pressure
10. a. If high desired inlet pressure: Open pressure regulator further until expander speed reaches 4000RPM. (Ensure pressure and flow sensors do not exceed their maximum ratings)

    b. Increase brake pressure using brake screw, to reduce speed to 500-1000RPM

    c. Repeat a & b until the required inlet pressure is attained. (This is done to avoid excessive expander speeds that can occur at low torque and high-pressure conditions)
11. Set the expander speed using brake screw to increase/decrease torque
12. Take a reading by clicking record button in the PicoLog software and the run code button in MATLAB
13. Save the PicoLog results to a CVS file
14. Alter the pressure/speed as described above and take any further readings required
15. Stop the expander by slowly reducing pressure regulator to the off position
16. Close the shutoff valves
17. Turn off the computer and power supply

6.4. Friction Testing Experimental Rig

The experimental test setup was modified to allow the friction losses within the Wankel expander to be measured, thus enabling a better comparison between the CFD and the experimental data. The modified friction test rig consisted of the following (see Figure 6.13):

- Static shaft Wankel expander
- Dualsky XM6360EA-10 Brushless DC motor
- Dualsky XC9036HV Electronic speed controller
- Laboratory power supply, set to 30V
- Arduino Mega 2560 board, with potentiometer.
- Datum torque meter
- Computer with Windows 7, Datum torque meter software, MATLAB and PicoLog6

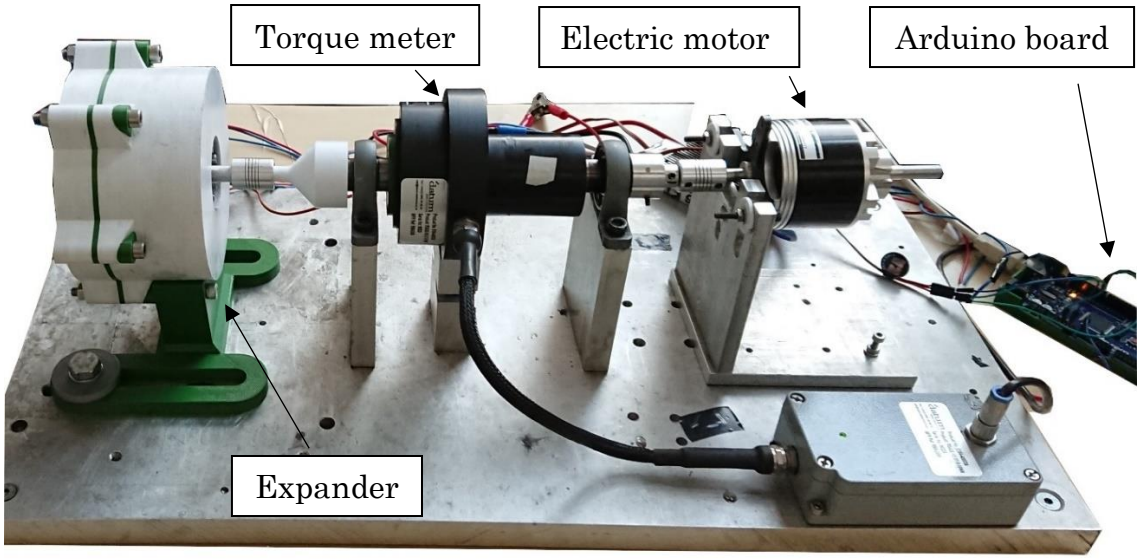


Figure 6.13 – Friction testing rig with brushless DC motor

6.4.1. Electric Motor

The laboratory power supply at 30V was used to power the brushless DC motor with a maximum current of 3A. To produce the input signal to the electronic speed controller of the motor, the Arduino board was utilised with a script obtained from [112]. A rotary potentiometer and voltage divider circuit provided a controllable 0-5V voltage input to the Arduino board. The script then created a pulse width modulated (PWM) signal, with a frequency proportional to the input voltage. The motor's electronic speed controller read the PWM signal and controlled the motor's power proportionally. This allowed the speed of the motor to be controlled via the potentiometer.

6.4.2. Torque Meter Calibration

Unfortunately, in this setup the load cell cannot be used to measure the torque as the electric motor is in its place. Therefore, the torque meter had to be utilised despite its lower accuracy. In order to read the most accurate results possible, the torque meter was calibrated against the loadcell measurements, when the rig was setup as in the previous section.

6.4.3. Alterations to Expander Assembly

To prevent the compression of gas affecting the measurements of the friction, one of the Housing side parts of the expander was rotated by 60°. This meant that at any point in rotation, all chambers of the expander were connected to at least one port (either inlet or outlet). As both the inlet and outlet pipes were disconnected, this connected all chambers directly with ambient conditions at

all rotations and therefore, very little compression could occur. Therefore, the majority of the measured power loss should be due to friction or other mechanical losses.

6.4.4. Test Procedure

1. Turn on the computer and power supply
2. Open Datum torque meter logging software
3. Set expander speed using potentiometer
4. Once speed is stable, start logging results on Datum logger
5. Stop logging after a given time (usually 20 seconds)
6. Repeat steps 3-5 for all speeds desired
7. Slowly reduce expander speed to zero using potentiometer
8. Turn off computer and power supply

6.5. Chapter Summary

- The design of an experimental test rig with the following capabilities was described:
    - Regulated compressed air inlet flow up to 1.2bar gauge pressure
    - Inlet and outlet gas pressure measurements up to 6.9bar absolute pressure
    - Inlet and outlet gas temperature measurements tested for the range 0-41°C
    - Inlet gas flow rate in the range of 2.5-25m^3/hour
    - Output shaft torque measurement up to 2.35Nm

    - Output shaft rotational speed measurement up to 20,000rpm
- The combination of sensors in the test rig allow the power output and isentropic efficiency of the expander to be calculated with total uncertainties of ±2.83W and ±0.526% respectively.
- A modified test rig for the measurement of friction losses within the Wankel expanders was also described.

CHAPTER 7

EXPERIMENTS: PROTOTYPE DESIGNS AND TEST RESULTS

7.1. Chapter Introduction

This chapter follows directly from the previous and details the design of each Wankel expander prototype and then presents and discusses the experimental test results. The design of each of the components within the prototypes is described and reasons given for the design choices. The problems experienced with the first prototype, which are seen in its results, and how these problems shaped the design of the second prototype are described.

7.2. First Wankel Expander Prototype (Mk1): Design

7.2.1. Description

An exploded CAD view for the first prototype designed of the static shaft Wankel expander is shown in Figure 7.1 whilst Figure 7.2 shows a cross-sectional view of it assembled. It should be noted that in all prototype designs there are no rotor side seals, sealing is instead achieved through a small clearance. In Figure 7.2, the yellow coloured components represent those that are stationary during operation. The red components rotate at 2/3 the speed of the blue components and the grey components are bearings. The stationary

components consist of the two halves of the outer casing and the stationary shaft. The red rotating components consist of the rotor and its internal gear. The blue rotating components consist of the rotor's housing, the two housing sides, the housing's external gear and the output shaft.

In this design, the rotor rests on the shaft via a ball bearing, whilst the shaft itself rests on two ball bearings connected to the housing sides. The shaft is clamped to the outer casing side A to prevent it from rotating. The housing assembly is supported by two ball bearings which rest on each of the outer casing sides. The inlet flow is connected to the protruding shaft end, which is hollow allowing the flow in. Both gears are bolted to their respective components with 3 bolts, M2.5 for the housing gear and M3 for the rotor gear. The output shaft fits into housing side b with an O-ring to seal and is then bolted on using six M3 bolts. There are also two O-rings around the periphery of the housing on both sides to seal against both housing sides.

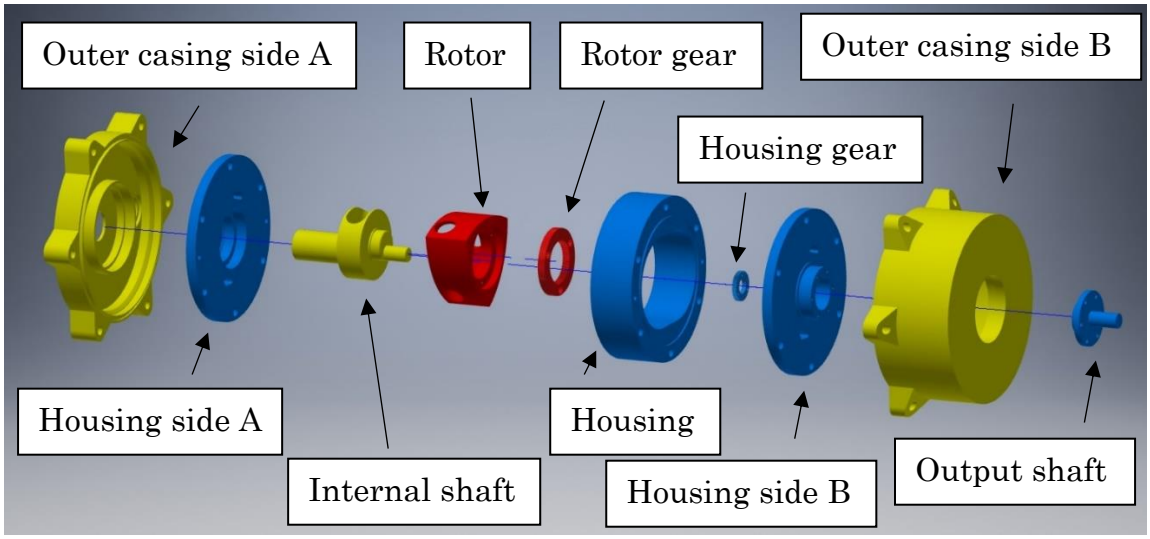


Figure 7.1 – Exploded CAD view of the static shaft Wankel expander's first prototype

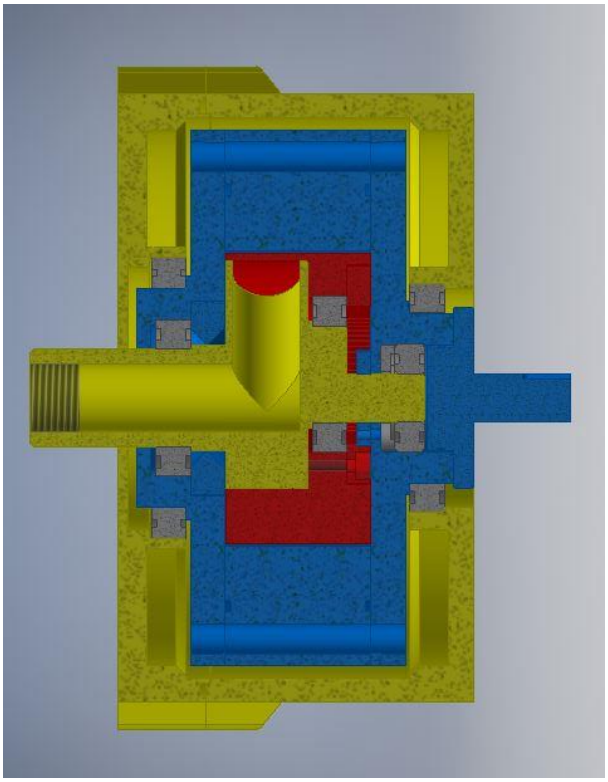


Figure 7.2 – Cross-sectional view of CAD assembly for the static shaft Wankel expander's first prototype

7.2.2. Materials

In order to help conceptualise the design, test the assembly process and check the clearances, the design was first 3D printed before it was sent for manufacturing. This was useful to ensure there were no major overlooked errors in the design before committing to the expensive and time-consuming manufacturing process. The 3D printed assembly was also useful in setting up the test rig whilst the manufacturing was taking place, the metal manufactured expander could then simply be swapped for the 3D printed one when it was complete, saving further time. Finally, the 3D printed assembly proved good enough to run some preliminary tests. Although it did not achieve the same performance as the metal expander assembly, the results weren't

that dissimilar. Figure 7.3 shows the photos of the 3D printed parts, the results for the tests are shown later in this chapter.

Figure 7.3 - 3D printed Plastic parts

The main problem the 3D printed parts faced during testing was plastic deformation of the side plates during operation. They ballooned out slightly under higher pressures, causing increased leakage between the rotor sides and housing sides drastically and permanently reducing the performance. The 3D printed parts also had more problems achieving the designed tolerances and the shaft part failed under high speed operation. However, it is worth further research into utilising plastic parts for some of the parts that experience less stress concentrations, such as the housing and the rotor, as it is cheaper and lighter.

The manufactured metal assembly was predominantly manufactured from Aluminium 7075 as it is easier to machine and therefore cheaper and quicker to manufacture. It also has very similar tensile and shear strength to carbon

steel. Aluminium 7075 also benefits from being more than half the density of carbon steel, which significantly reduces the frictional losses in the bearings of the moving parts. This is the main reason all rotating components (Housing, housing sides, rotor and output shaft), shown in Figure 7.4 (top row and bottom left) were manufactured from aluminium 7075.

Figure 7.4 -Metal 1st prototype parts, Steel (Internal shaft, bottom-right) and Aluminium 7075 (the rest)

The internal shaft, shown in Figure 7.4 (bottom right), was the only component that was manufactured from carbon steel, this was because carbon steel has higher fatigue strength, elastic modulus and shear modulus. This is desired because the shaft experiences both the highest stress concentrations and has them reversing at very fast rates due to the high operational speed.

This significantly lowers its fatigue life. The shaft also contributes most to the positioning of the rotor within the housing, and with such small designed clearances, even small deformation could cause unwanted contact between moving parts. However, as the shaft weight rests on bearings, future research should be done to assess the use of a lighter weight material to reduce bearing friction.

7.2.3. Gears

The Wankel expander utilises two gears, an internal and an external, to assist the required motion of the rotor within the housing. As previously mentioned in the literature review chapter, the gears do not experience significant torque and are primarily there for locating the rotor rather than power transmission. This fact made the design of the gears much easier and only the following limitations had to be considered:

- A speed (tooth) ratio of 3:2
- A centre distance equal to the Wankel expander's eccentricity (4.8mm).
- Internal gear: Maximum outer diameter of 48mm, in order to fit into the rotor.
- Internal gear: Enough space for bolt holes between the outer diameter and the teeth.
- Internal gear: Maximum face width of 3mm to allow room for bearing and inlet port.
- External gear: Minimum bore of 10mm to fit the shaft through it.

- External gear: Enough space for bolt holes between bore and the teeth.
- External gear: Same face width or smaller than internal gear.

As we had no facilities to manufacture gears, it was decided to buy a gear pair of a certain size and modify them for the Wankel expander. A pair of steel gears were chosen because they were available with a smaller tooth module. A smaller tooth module was desirable for two reasons, the first is that a smaller tooth module has more teeth and therefore less backlash. This is beneficial for the Wankel expander as it means the positioning is more accurate and therefore clearances can be smaller. The second reason is that a smaller module means that each gear tooth has a smaller tooth depth, which allowed more metal between the teeth and the outer diameter (internal gear) or inner bore (external gear). This was useful to allow more space for bolt holes.

Therefore, gear pairs were first narrowed down by tooth ratio and centre distance, and then the pair with the smallest module was selected. Once the gears were received, they were modified in the following ways:

- **Both** gears were machined to the correct face width
- **Both** gears had three bolt holes drilled in the correct locations
- The outer diameter of the **internal** gear was machined to the correct size
- The bore of the **external** gear drilled to the correct size

7.2.4. Bearings

Bearing choice provides a difficult challenge in the static shaft Wankel expander due the space restrictions within the device. Ideally much more research could be focussed on development of this area, however, this design was the first prototype, who's primary objective was proof of concept and validation of fluid flow simulations. Therefore, ball bearings were chosen mainly for their thin profile, which allowed them to fit in the available space in the design. The ball bearings were all chosen to have maximum operating speeds above 10,000rpm and maximum loads of at least double their calculated carrying loads. The choice of bearings is given in Table 7.1.

All the bearings were also chosen to be pre-greased and fitted with rubber grease seals as there was no designed lubrication system. However, during the experiments, it was found that due to the gas pressure differentials constantly changing within the expander, the grease seals of some bearings were being forced into the bearing's balls, causing excessive friction. Therefore, for the remainder of the experiments, the grease seals were removed, which did allow the loss of grease over time and would need to be addressed in a commercial design.

Table 7.1 – Bearing specifications

Bearing	**Specification**
Small internal shaft bearing	SKF 619002RS1
Large internal shaft bearing	SKF 618042RS1
Small outer housing bearing	Budget 68072RS
Large outer housing bearing	Budget 68092RS
Rotor bearing	SKF 61804-2RSI

7.2.5. Clearances and Tolerances

There are multiple clearances between the moving parts of the static shaft Wankel expander, and the accumulation of part tolerances had to be considered when designed. The places in which the clearances were most important is the clearance between the rotor and housing sides, between the rotor apexes and housing, and between the rotor inner bore and the shaft's inlet port diameter. These clearances are crucial as they all need to be as small as possible in order to reduce leakages but need to be kept from contacting to prevent unwanted friction and wear. As a machining tolerance of ±0.025mm was claimed to be achievable by the manufacturers, the clearance between the side housing and rotor side and the clearance between the rotor bore and shaft were designed to be 0.05mm. The clearance between the apexes and the housing had more uncertainty, as it relied on the machining tolerance of both parts, the cumulative location accuracy of three

bearings, the gears' backlash, and the gears' fitting location accuracy. Due to these factors, this clearance was designed to be a larger value of 0.15mm.

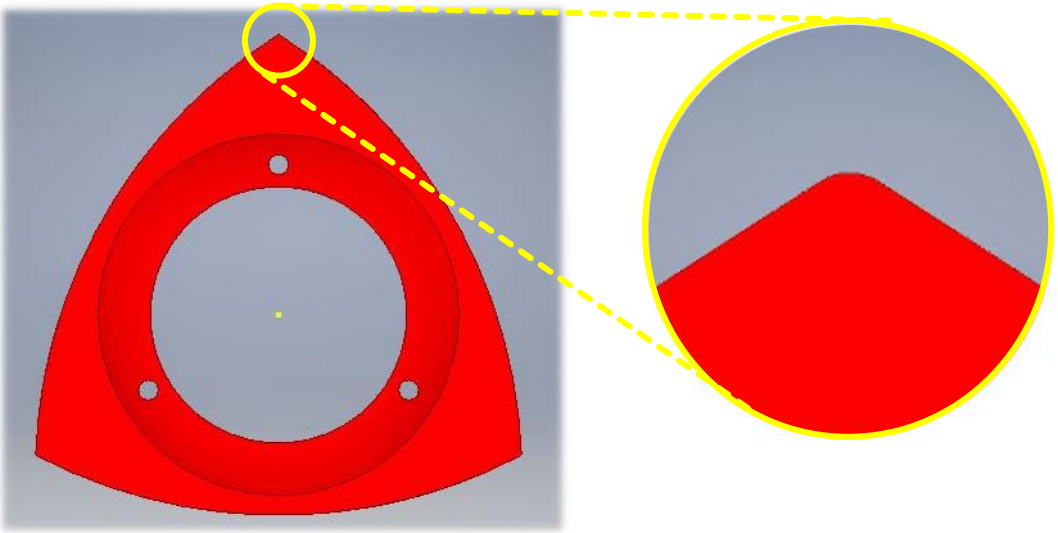


Figure 7.5 – Curved Apex on manufactured rotor

Due to the CNC milling manufacturing method, the claimed tolerance could not be guaranteed at the rotor apexes if they were designed as points. Therefore, the design was altered to have small curved apexes with radii of 0.25mm, Figure 7.5 demonstrates this.

7.2.6. Sealing

To prevent high pressure inlet gases escaping out of the expander, an O-ring was utilised between the housing and each of the housing sides. O-rings are utilised in this case as there is zero relative movement between these parts, housing O-ring grooves can be seen in Figure 7.3 and Figure 7.4. Furthermore, a smaller O-ring was utilised on the diameter of the inserted section of the output shaft. This left only one remaining route for high

pressure gases to escape the expander housing assembly (other than through the expander as desired) which was the gap between the shaft and one of the housing sides. This was originally designed to have a rubber shaft seal to prevent gas leakage, however, in initial experiments, it was found that the seal produced too much friction, and the expander performed significantly better without it. This could be improved in future designs by using a lower friction seal material or by utilising an oil lubrication system.

7.2.7. Inlet and Outlet Ports

The outlet ports were simply designed with two requirements. The first was that they should be open when the rotor reaches maximum volume position and should close again when the rotor reaches minimum volume position. The second was that they should be as large as possible whilst maintaining the first requirement. The resultant shape is presented in Figure 7.6, where the yellow dashed line was the shape designed from maximum and minimum volume positions of the rotor. To maximise the port flow area, the outlet ports were mirrored in both side plates, this also helped keep the flow and forces symmetrical, reducing the chance the rotor would be pushed one way against a housing side. Finally, to machine the outlet port shape, the corners had to be curved with a radius greater than that of the machine tool (1.5mm in this case), so a radius of 1.6mm was used. The solid line in Figure 7.6 represents the final curved port design.

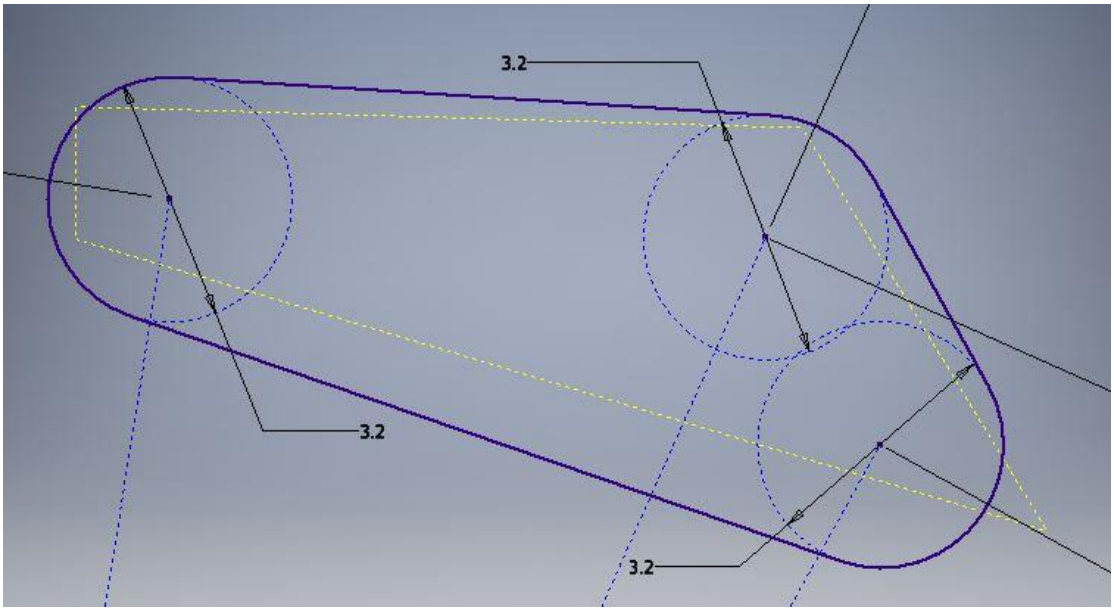


Figure 7.6 – Outlet port design, yellow dashed line was original shape

The inlet air flow for the static shaft Wankel expander is routed through the shaft and then through the ports in the rotor flanks. The inlet ports in the rotor were designed simply as straight drilled holes, with the same diameter as the shaft. In the original CFD simulations, the rotor ports were designed to be in the centres of the rotor flanks, as this is where the rotor is thinnest. This produced a smaller clearance volume for the expander, as less volume is added to the expansion chamber, which in turn increases the volume ratio. However, for the first prototype design the ports were moved further to the leading edges of the rotor flanks. This allows the inlet flow to be directed straight to the largest section of the chamber volume, theoretically reducing flow restrictions.

7.2.8. 3D Printed Outer Casing

As mentioned earlier, 6.3.10 some parts were 3D printed, this was the case for the two halves of the outer casing. The case could be 3D printed because

its strength, surface finish and manufacturing tolerance were not critical to the performance of the Wankel expander. However, one important design element of the casing was that it should have enough space around the outside of the expander housing and housing sides to allow the outlet flow to proceed to the outlet port with little restriction. The gap can be seen in the assembly cross-section in Figure 7.2.

7.2.9. Inlet Manifold/Shaft Flow Analysis

As the experimental inlet sensors were located before the internal shaft of the Wankel expander and the previous chapter's CFD simulation work involved simulations of just the internal parts of the Wankel expander, a separate steady state CFD simulation was carried out on the flow inside the shaft. This allowed an estimation of how the flow is altered between the sensors and the internal part of the expander. This simulation used an inlet mass flow rate of 0.01kg/s as this was the maximum observed in CFD and a fluid temperature of 300K. The outlet boundary condition was set to the desired pressure for the inlet to the Wankel expander's chambers. Therefore, the simulation could be used to find both the pressure drop over the shaft length and the required shaft inlet pressure for a desired internal expander inlet pressure.

For the pressure contour results shown in Figure 7.7 the gauge pressure at the outlet was set as 6bar, as this would produce the largest pressure drop and therefore the worst-case scenario. The pressure drop between the inlet and outlet of this simulation is approximately 0.006bar and the required inlet

gauge pressure is about 6.0065bar. It was originally planned to adjust the experimental inlet pressure readings based on CFD results, however, as the pressure drop is less than the accuracy of the pressure sensors, this was not done. It is also noticed that there is a large pressure concentration at the edge of the right-angle corner. Therefore, this is the greatest point of efficiency loss, which is to be expected as the right-angle is not an aerodynamic shape. The shaft was initially designed in this way to keep manufacturing time and cost low. However, the right angle was altered in the second prototype to have a curved outer profile.

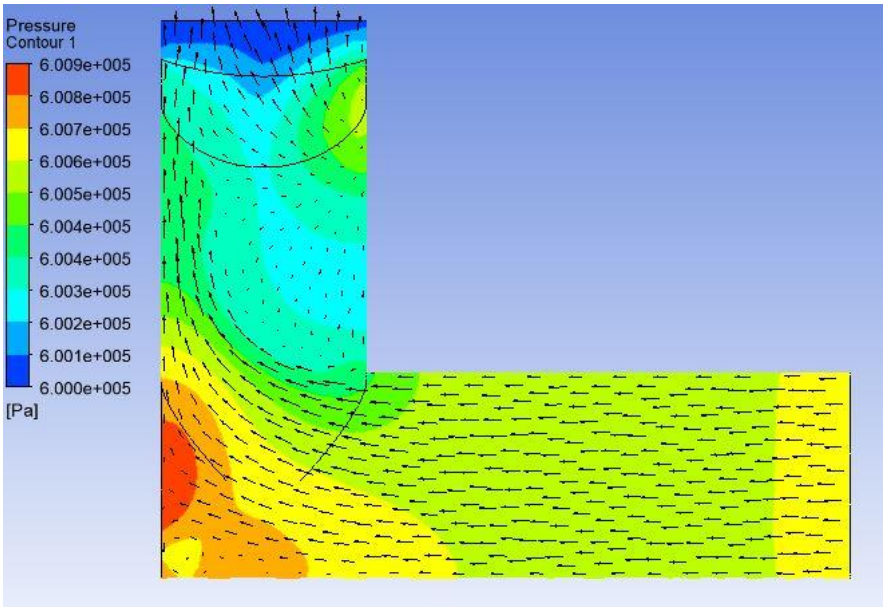


Figure 7.7 – CFD analysis of the flow through the inlet shaft (flow pressure contours and velocity vectors)

7.2.10. FEA of Shaft

To ensure the design of the shaft was strong enough not to fail or undergo excessive bending during operation, static structural finite element analysis (FEA) was performed. A static simulation was performed as the shaft remains

stationary. Even though the forces it experiences are constantly changing, the maximum force can be estimated as the normal pressure force on the rotor flank when presented to the inlet pressure. This force was calculated at a 6bar inlet pressure as 1024N on the rotor, to test the shaft at the maximum force conditions. The force was located at the shaft's rotor bearing face, which supports the forces of the rotor. The other two bearing faces on the shaft, which are for each of the housing side bearings, were defined as fixed supports.

Figures 7.8 and 7.9 shows the stress distribution and deformation of the internal shaft under maximum load. The maximum von-Mises stress is calculated as 67.947MPa, as the internal shaft is manufactured from structural carbon steel, which has a yield strength of approximately 250MPa and endurance limit of approximately 270MPa, the shaft should never experience plastic deformation and should last an infinite number of cycles, which is required as the speed of the expander means it will undergo a large number of cycles in a relatively short space of time. The location of the maximum stress is shown to be at the edge between the small diameter section of shaft and the rotor section. This is useful to know if future design improvements involve changing the shaft to a lighter but weaker material. The stress concentration at this area could be reduced by adding a fillet, chamfer, or grooves.

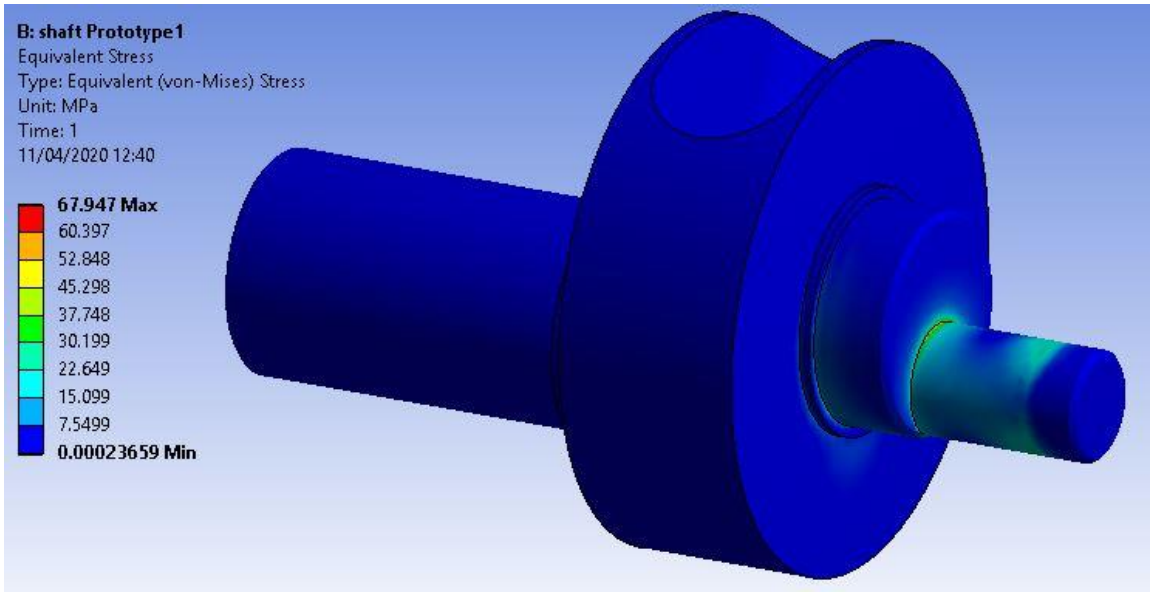


Figure 7.8 -Stress distribution results for FEA of internal shaft

The maximum deformation of the shaft occurs at the shafts interface with the rotor. This could cause problems due to contact with the rotor or contact between the connected rotor the housing. However, with the current material, the maximum deformation is approximately 0.0026mm, which would not be enough to close the apex clearance gap of 0.15mm.

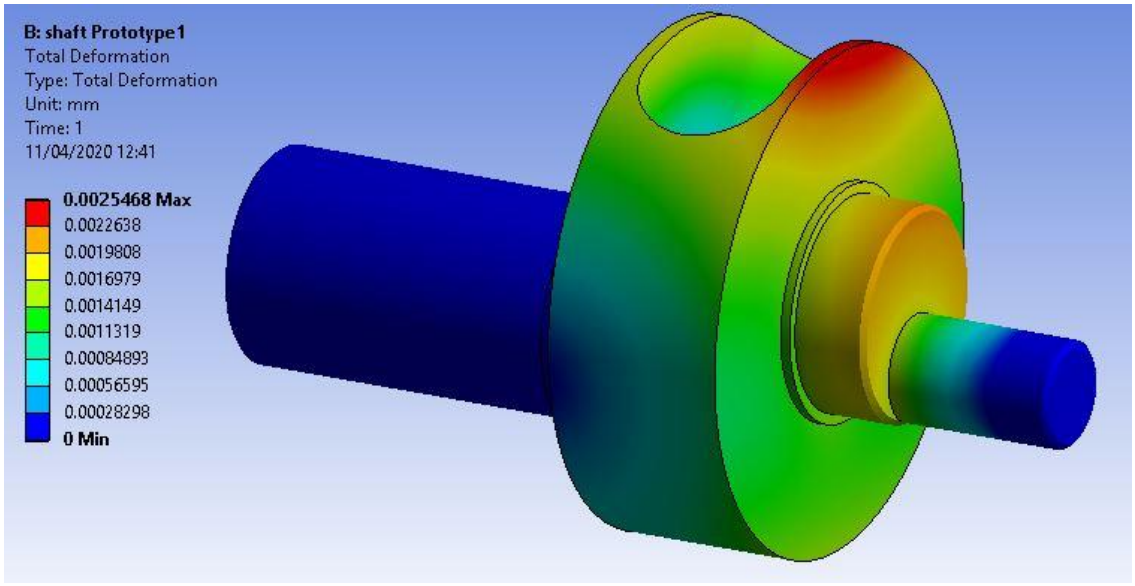


Figure 7.9 – Deformation of the internal shaft due to applied forces

7.2.11. FEA of Housing Sides

Structural FEA was also carried out on both housing sides. The plastic 3D printed housing sides caused large drops in performance due to 'ballooning' deformation causing additional leakage past the rotor sides. Therefore, the main purpose of this analysis was to check if the estimated maximum deformation of the aluminium housing sides is acceptable. The force was applied to the housing sides as a 6bar pressure applied at the area where the expansion chamber would be located at the maximum inlet pressure (i.e. immediately after the inlet port opens).

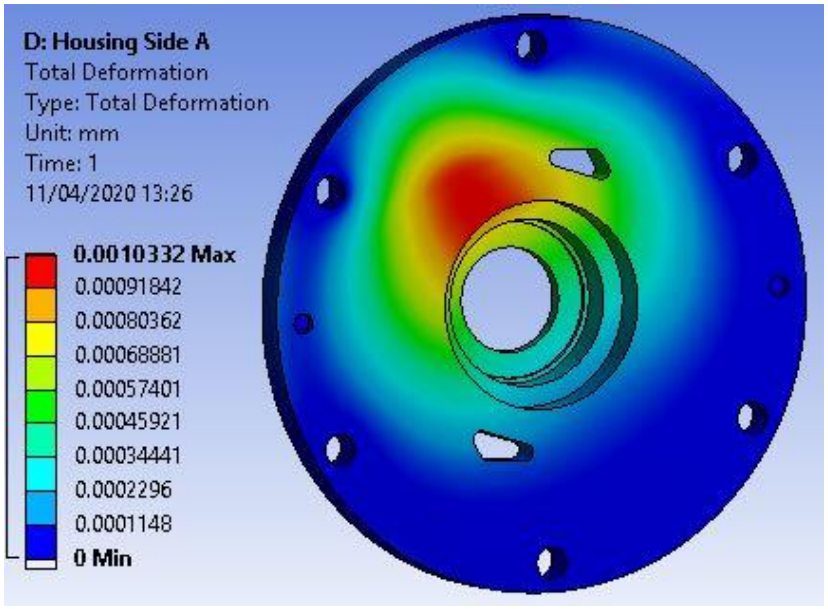


Figure 7.10 – FEA deformation results for housing side A

Figures 7.10 and 7.11 show the FEA deformation results for housing sides A and B respectively, in which the maximum deformations are 0.00103mm and 0.00098mm. If this deformation occurred during operation, the designed axial gap between the rotor and a housing side of 0.05mm would be increased to 0.05103mm in the area most affected. This increase is not thought to be

significant and additionally, it would be much less at most of the other lower inlet pressures conditions. The maximum von-Mises stress found was 4.34MPa which is far below the yield strength of most metals and therefore no further consideration is needed in that regard.

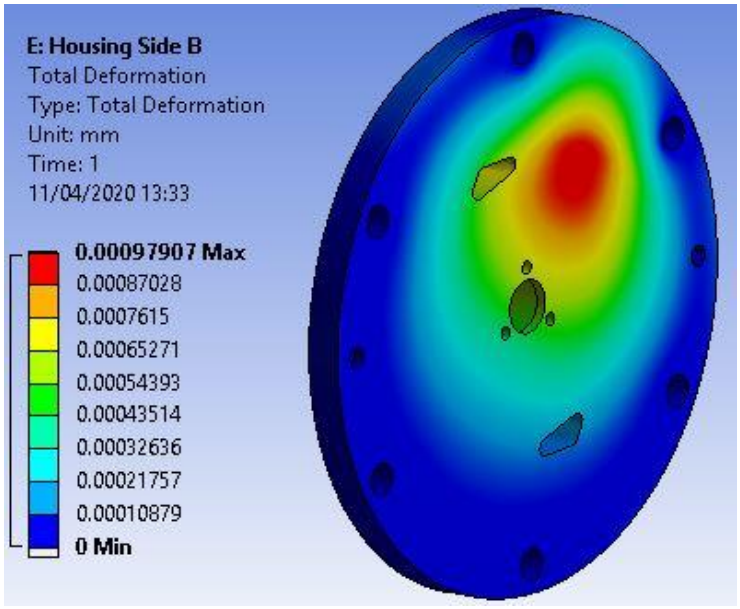


Figure 7.11 – FEA deformation results for housing side B

7.2.12. FEA of Output Shaft

Structural FEA was also performed on the output shaft, shown in Figure 7.12. The model was simply defined by applying the maximum torque the CFD calculated to the shaft. The bolt holes were defined as fixed supports. The maximum von-Mises stress was found to be 310.09MPa, which is below the yield limit of the material (aluminium 7075), however, as aluminium does not have an endurance limit, it is predicted from the S-N curve for the material that it would fail after 300,000 cycles at these conditions. Although, this is not likely to be the case, as the maximum operating conditions are unlikely to be experienced for significant time (if at all) during testing, the design

should be improved in future to include structural features such as a chamfer, fillet or step, which were all initially left out to save time and cost during manufacture.

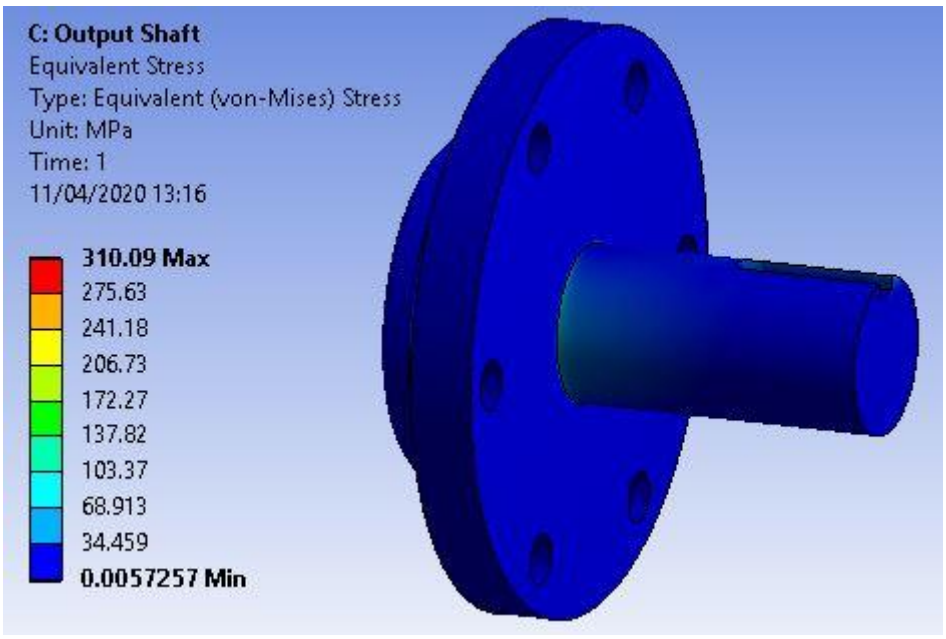


Figure 7.12 -FEA stress distribution results for the output shaft

7.2.13. Measurement of Apex/Side Clearances after Manufacture

Once the assembly had been manufactured, the clearances were found to deviate slightly from the design. The clearances between the rotor apexes and the housing were measured with steel shims of known thickness and were found to be approximately 0.2mm instead of the designed 0.15mm, therefore the value of 0.2mm was used in the CFD related simulations.

7.2.14. Static Shaft Wankel Expander Assembly Method

1. Screw external gear to housing side B, using three M2.5 bolts
2. Screw internal gear to rotor, using three M3 bolts
3. Insert O-rings into housing O-ring grooves

4. Insert locating dowels into housing
5. Insert bearing spacer and then the bearing into housing side B
6. Insert bearing into housing side A
7. Connect the housing and the housing side B parts, aligning the dowels with holes
8. Press the bearing onto the internal shaft.
9. Press the rotor onto the internal shaft over the bearing, check free rotation
10. Insert small end of internal shaft into the bearing in housing side B
11. Check rotor and shaft can rotate freely and that gears mesh
12. Place Housing side A over the large end of the external shaft, pressing the shaft onto the gear and aligning the housing dowels with the dowel holes.
13. Bolt Housing Sides onto housing using six M6 bolts and nuts
14. Check free rotation of shaft and rotor, if not tap lightly with a wooden mallet and check again
15. Press outer bearings and spacers onto Housing sides A and B
16. Press both outer casing halves onto assembly and secure with six M6 bolts.

7.3. First Wankel Expander Prototype (Mk1): Results

During the experimental testing of the first prototype, continual assessment and improvement was being made to the experimental rig, the sensors, the expander assembly procedure, and test procedures. Due to this, some of the

following results have a large scatter, some have less data points and some have restricted operating conditions. Precise control of the inlet pressure proved impossible, with fluctuations from the compressed air supply upstream affecting the expander's inlet pressure. The following results have therefore been grouped into inlet pressure groups to allow comparison, where the inlet pressures were rounded to the nearest 0.1bar, this adds further scatter in the results. Finally, the following results (mk1) do not include flow rate or efficiency, because an accurate method of measuring the inlet mass flow rate was not found until late into the testing of the second prototype. However, the power output results still help illustrate the development of the expander and the problems faced.

7.3.1. Plastic 3D Printed

Initial tests were attempted using the 3D printed plastic assembly, before the metal expander was manufactured. However, it was found that the two housing sides were not strong enough to withstand the internal pressure forces and plastically deformed. The plastic deformation lead to significant leakage when tested above 0.3bar inlet gauge pressure. This pressure is much too low for most applications and therefore the expander with 3D printed plastic sides is not viable. However, in a few of the following tests the plastic shaft and the plastic rotor were utilised with some success.

7.3.2. Metal Expander

Initial testing with the metal expander assembly (Steel internal shaft and Aluminium 7075 for all other parts) displayed problems with excessive friction, especially when the inlet pressure was raised. Looking at Figure 7.13, the initial manufactured assembly (unmodified) shows its highest power output at 0.5bar inlet pressure, with lower power outputs at 0.3bar and 0.7bar. The reason for a lower power output at 0.7bar is the increased friction, and when the inlet pressure was raised above 0.8bar, the friction became large enough to overcome the generated power and the expander would stop rotating.

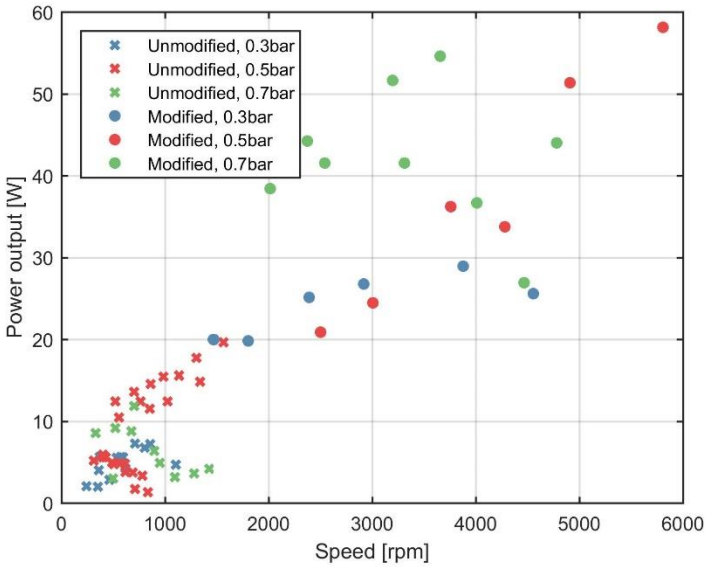


Figure 7.13 – Power output against speed for different inlet pressures (unmodified and modified 1st prototype, metal expander)

After further investigation, friction marks were found on the housing sides, but only at certain positions. Friction marks were also found on the rotor sides and around the rotor's internal bore where it interfaces with the internal

shaft. Furthermore, less noticeable friction marks were also found on certain areas of the shaft. It was reasoned that the pressure force from the inlet flow was creating a force on the rotor internal surface and that the single ball bearing supporting the rotor was not strong enough to prevent radial rotation between the two parts. Figure 7.14 shows the location of the pressure force, the centre of the rotor bearing which acts as the pivot point, the resultant rotation torque, and the areas where friction marks were found (pink lines).

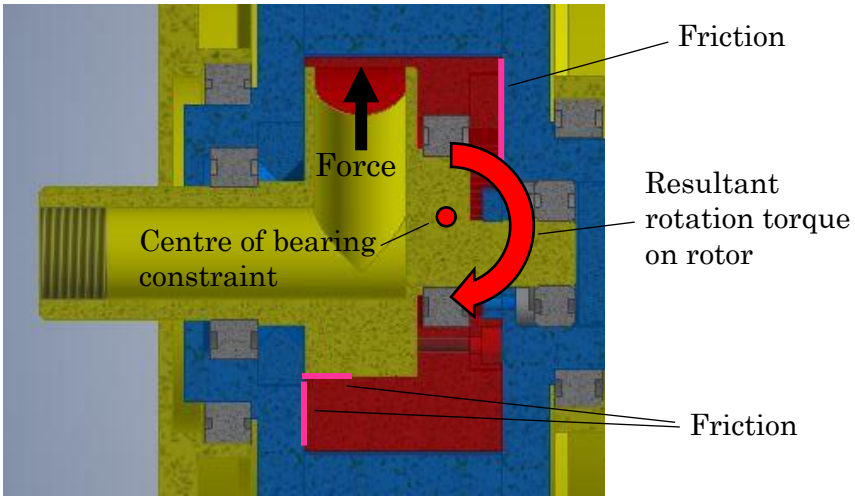


Figure 7.14 – Pressure force causing friction between rotor and shaft

In an attempt to prevent this friction occurring and improve the performance of the expander, the internal shaft was modified. This modification consisted of four ports being drilled into one face of the internal shaft and those ports being joined to the high-pressure inlet flow (See Figure 7.15). In theory, this would provide a second force from the gas pressure on the rotor, causing a rotational moment in the opposing direction. Furthermore, as the original problematic rotational moment was also cause by the inlet pressure, this solution should work across all inlet pressures. The ports were designed at a

size and perpendicular distance from the pivot point to theoretically cancel any moments on the rotor as demonstrated in Figure 7.16.

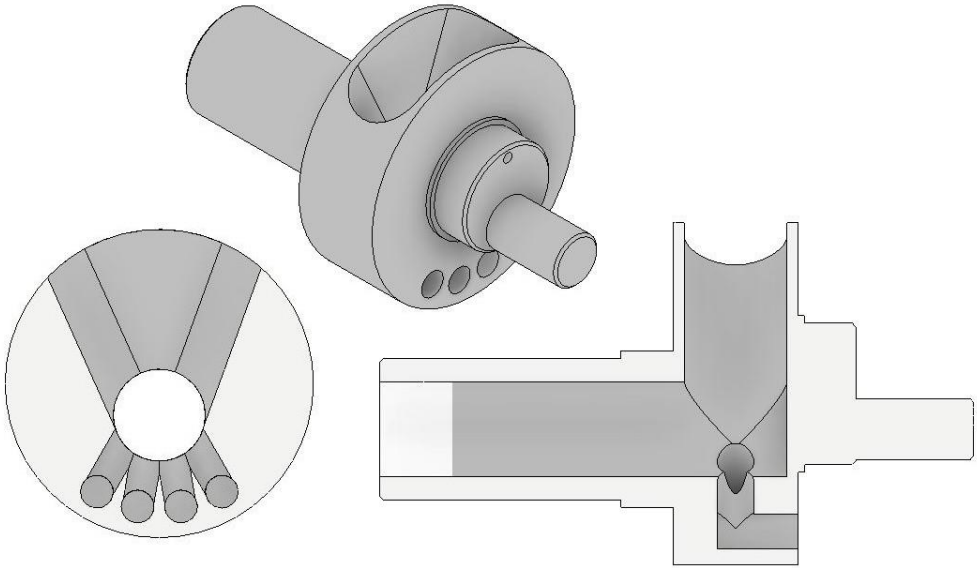


Figure 7.15 – Modified 1st prototype internal shaft with section views

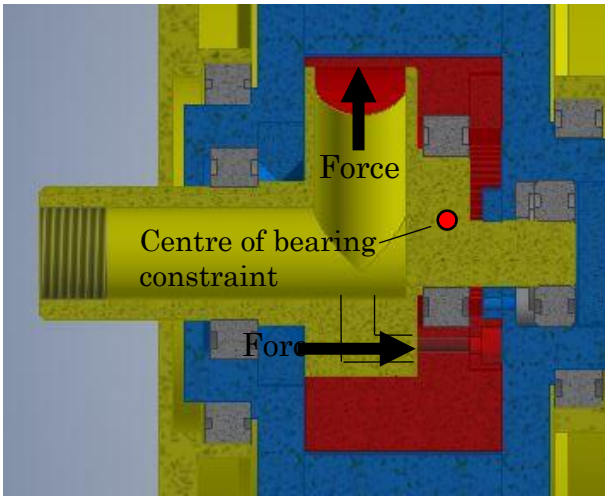


Figure 7.16 – Pressure force moment balance on rotor

Figure 7.13 also shows the power output results when the expander was tested with the modified shaft. There is a clear improvement in both power and top speed of the expander for all the tested inlet pressures. However, the 0.7bar inlet pressure still appears to have little increase in power compared

to the 0.5bar inlet pressure, in fact some 0.5bar cases outperform it. This is due to two different reasons, the first being that the moment on the rotor cannot be truly balanced dynamically. This is because the four new ports provide an almost constant pressure force on the rotor, whereas the pressure force from the main shaft port on the rotor varies every time a rotor inlet port passes it. The second reason is that the four new shaft ports cause an axial force on the rotor, this force acts to drive the rotor and the shaft apart axially, which at high enough pressures can overcome the press fitting of the rotor bearing and force the rotor axially into the housing side. Additionally, even if the rotor bearing is more strongly secured, the increase in axial force will cause high friction in the ball bearing. Therefore, the 1st prototype, even with the modification, was not a viable design.

7.3.3. Apex Seals

As the CFD chapter included some cases with apex seals, a rotor was designed and manufactured, that had a slot in each apex for an apex seal. The rotor design with the designed apex seals is shown in Figure 7.17. The apex seals each had an accompanying leaf spring to press them into the housing surface. The rotor was manufactured from the same material as the rotor without apex seals (Aluminium 7075) and the apex seals were machined from low friction Nylon 6. This was to reduce the friction and wear that would occur on the aluminium housing otherwise.

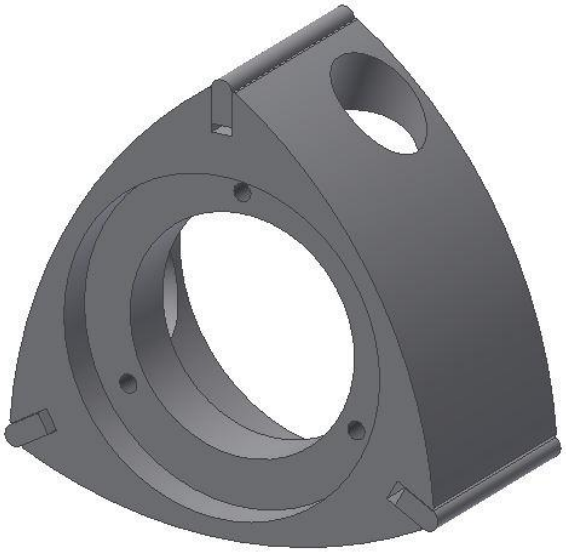


Figure 7.17 – Rotor design with apex seals

Figure 7.18 shows the power output results comparing the expander with and without apex seals. The modified shaft was used for these results. In all cases, the use of apex seals results in a decreased power output. This is attributed to the friction caused by the apex seals reducing the power significantly more than the reduction in leakage improves the power. This agrees with the results in the previous CFD chapter, where the use of apex seals only slightly improved the power output, when friction was not included. The previous CFD showed the main advantage of apex seals was a large increase in efficiency. Unfortunately, as an accurate flow meter was not available during these tests, the experimental efficiency is unknown and therefore testing the efficiency the expander with apex seals could be the focus of future research.

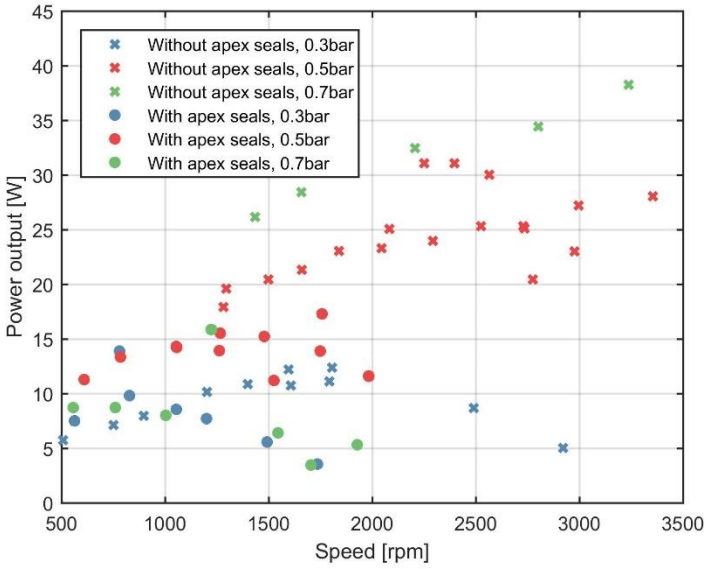


Figure 7.18 - Power output against speed for different inlet pressures (modified 1st prototype with and without apex seals, plastic shaft, other parts metal)

7.3.4. Friction Tests

The results from friction tests, carried out using the friction testing rig setup described in the previous chapter are shown in Figure 7.19. The results show an exponentially increasing power loss due to friction, giving a 2nd order polynomial trendline. The exponential increase in power loss corresponds to a linear increase of torque loss with speed, meaning the frictional forces inside the expander are also increasing linearly with speed. This increase is thought to be attributed mainly to the bearings and gears. This is because theoretically, no other friction source should be affected by speed. The trendline equation for the friction, shown in Figure 7.19, is used in the following section to adjust the CFD results to better compare them with the experimental results.

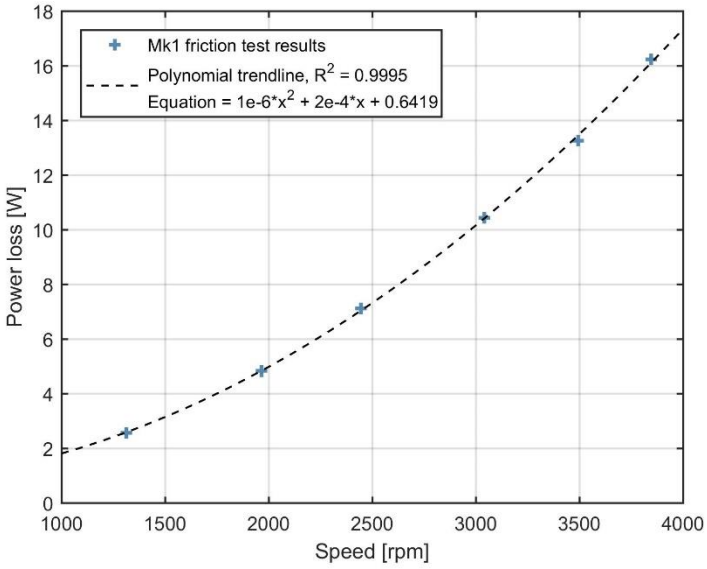


Figure 7.19 – Friction power loss test results for 1st prototype

7.3.5. Comparison to CFD

As mentioned previously, during the 1st prototype experimental tests, there was no accurate way of measuring inlet flow, therefore, the following only compare the power output results of the CFD and experimental. Figure 7.20, Figure 7.21 and Figure 7.22 show the comparison of result for 0.3bar, 0.5bar and 0.8bar inlet gauge pressures respectively. The first point to notice is that the friction adjustment on the CFD has a much larger effect for the low inlet pressure case. This is because pressure influences the power output but not the friction adjustment, therefore for higher pressures there is relatively less contribution from friction. However, the frictional power loss determined previously was measured without the expander running and therefore no gas pressure contributions to friction are included.

The second point is that the CFD results appear to become less accurate at larger inlet pressures. This could be due to multiple reasons, however, as the

previous sections mentioned, problems with friction were encountered during these tests. Specifically, the pressure force from the inlet forced the rotor to contact the internal shaft and the housing at points. Therefore, it can be logically concluded that the increased discrepancy in results at higher inlet pressures is due to this frictional problem. As this issue presented such a big problem with the validation of the CFD, it was decided to create a second design to address this problem specifically.

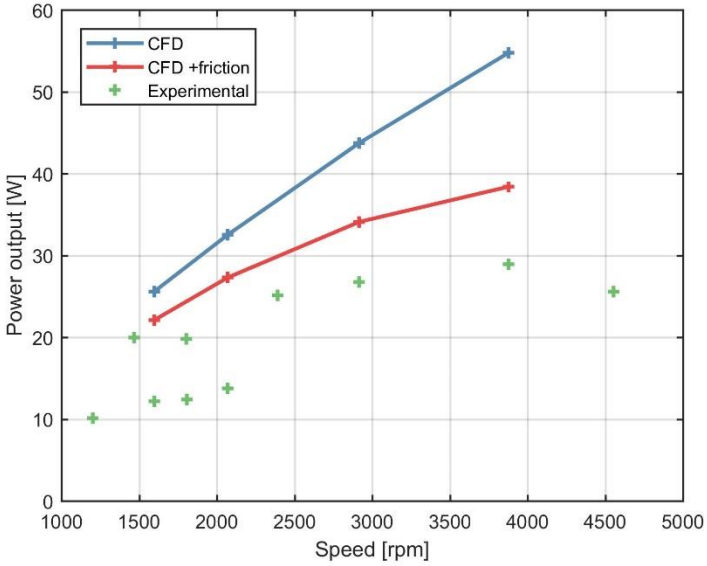


Figure 7.20 – Power output against speed results (0.3bar inlet gauge pressure, 1st prototype)

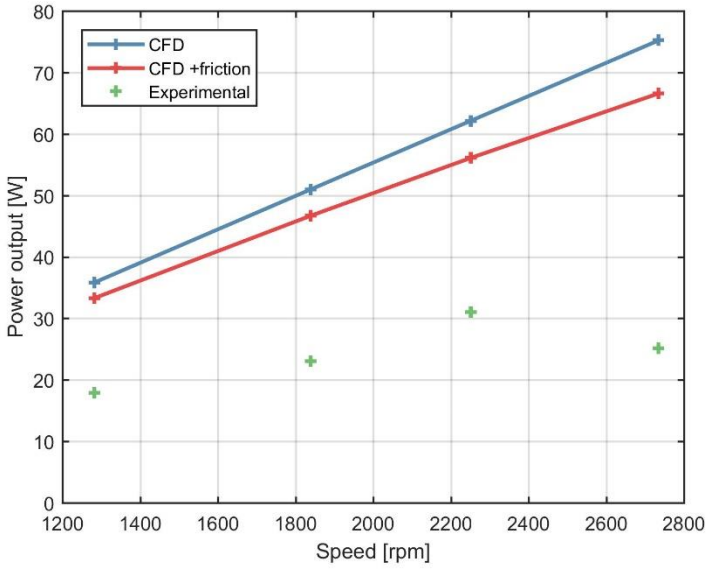


Figure 7.21 - Power output against speed results (0.5bar inlet gauge pressure, 1st prototype)

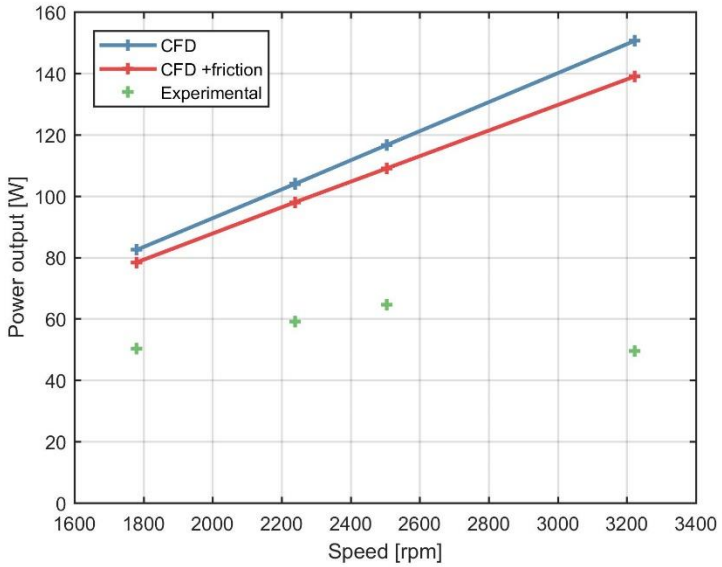


Figure 7.22 - Power output against speed results (0.8bar inlet gauge pressure, 1st prototype)

7.4. Second Wankel Expander Prototype (Mk2): Design

The second prototype was designed to address the inlet pressure related frictional problems of the first prototype. However, to save on design time, manufacturing time and cost, it was decided that the new prototype would use the same housing, housing sides and output shaft components as the first

prototype. This meant that any design changes of the rotor and shaft parts would have to fit within the predefined housing size and work with the same shaft-to-housing bearings. Figure 7.23 shows a comparison between the first and second prototype designs in section views and Figure 7.24 shows the manufactured second prototype rotor and internal shaft.

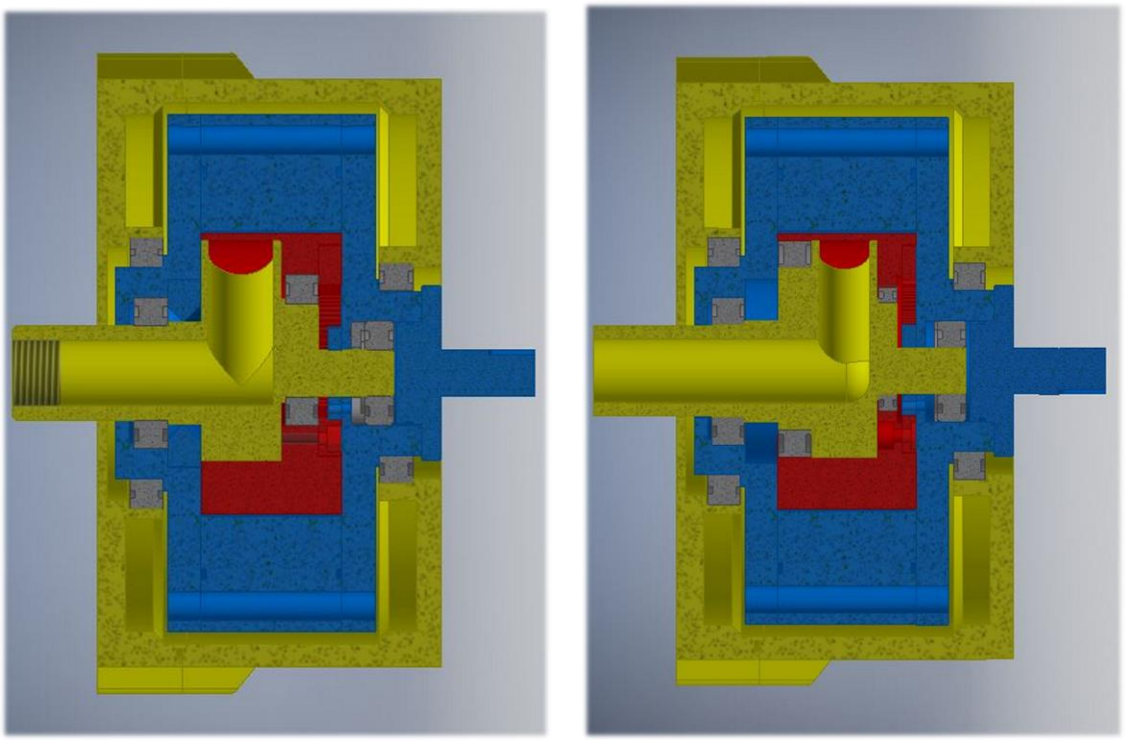


Figure 7.23 – First prototype (left) and second prototype (right) CAD assemblies

7.4.1. Rotor/Internal Shaft Assembly Changes

The primary change to the first prototype design is the addition of a second bearing between the internal shaft and rotor. The two bearings prevent the radial twisting of the rotor due to the inlet pressure, which was causing the contact between parts and therefore friction. In order to fit this additional bearing into the space provided, multiple other changes had to be made to the design. First, the original bearing between the rotor and shaft was reduced in

size as the width had to be reduced to create space. This resulted in a reduced load capacity of this bearing, however as the load is now shared between two bearings, the load it experiences is halved so it was not a problem.

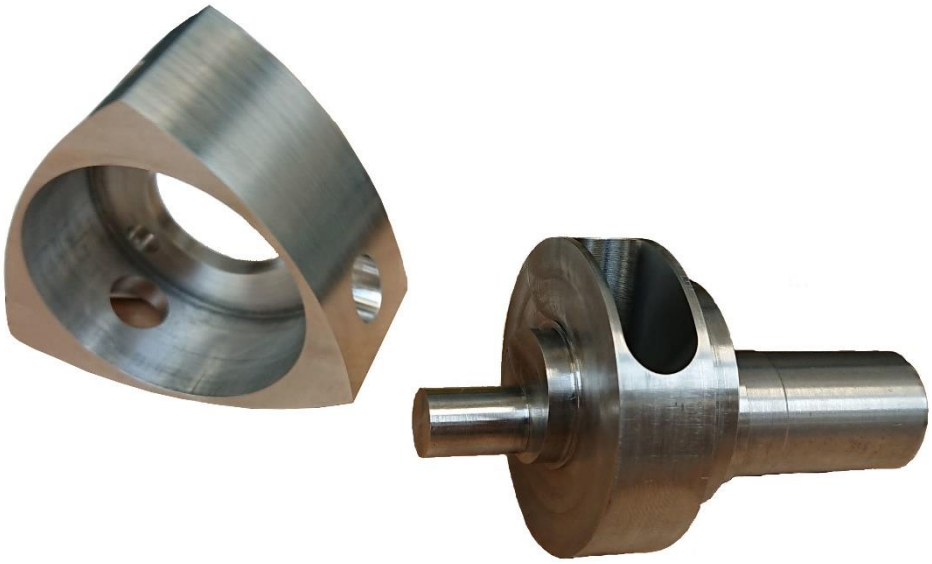


Figure 7.24 – Second prototype rotor (Aluminium 7075) and internal shaft (Steel)

Secondly, the internal gear mounted on the rotor also had to be reduced in width. However, previously the internal gear had bolt holes with counterbores for the bolt heads whilst the rotor holes were threaded for the bolts. Due to the gear's reduced width, there was no space for the counterbores, so the bolts were flipped in direction, and the gear holes were now threaded and counterbores were added to the rotor bolt holes. The downside to this is the shaft/rotor assembly must be disassembled to remove the gear.

The new second bearing had to have an outer diameter larger than the diameter of the shaft where the inlet port is located. This is to keep assembly possible as the shaft is inserted into the rotor from one side and the shaft diameter where the port is located must pass through the bore for the new

second bearing. This means the new bearing is significantly larger than the bearing in the previous prototype, plus there are two bearings for the same load. Therefore, the design has a load capacity much greater than required between the rotor and shaft, which means there is additional, unneeded bearing friction that could possibly be reduced with a different design.

The final change due to the added bearing was the reduction in size of the shaft inlet port and the corresponding reduction in diameter of the rotor's inlet ports. This unfortunately reduces the maximum inlet flow rate and therefore reduces the maximum speed of the device. This is a further reason the expander could benefit from a complete redesign in future research.

A small secondary change to the shaft is the creation of a curved edge where there was a sharp right angle in the first prototype. This is located inside the shaft's bore that the inlet fluid flows through. As noted in the design section of the first prototype design, the CFD of the inlet flow through the shaft showed a high-pressure zone and therefore efficiency loss at the right-angle edge. The curved edge was added to the second design with the hope that it would improve the flow capabilities of the shaft. Figure 7.25 shows the resultant pressure contours from a steady state CFD model, again with 0.01kg/s inlet mass flow rate and a 6bar gauge pressure outlet. The 0.017bar pressure drop observed in the CFD is greater than for the previous prototype (0.0006bar), although this is mainly due to the decreased diameter of the rotor port. Consequently, it is not possible to directly compared the two shaft flows in terms of pressure drop, however the new shaft with the curve corner shows

much better flow streamlines and smaller turbulent vortices which suggests more efficient flow characteristics.

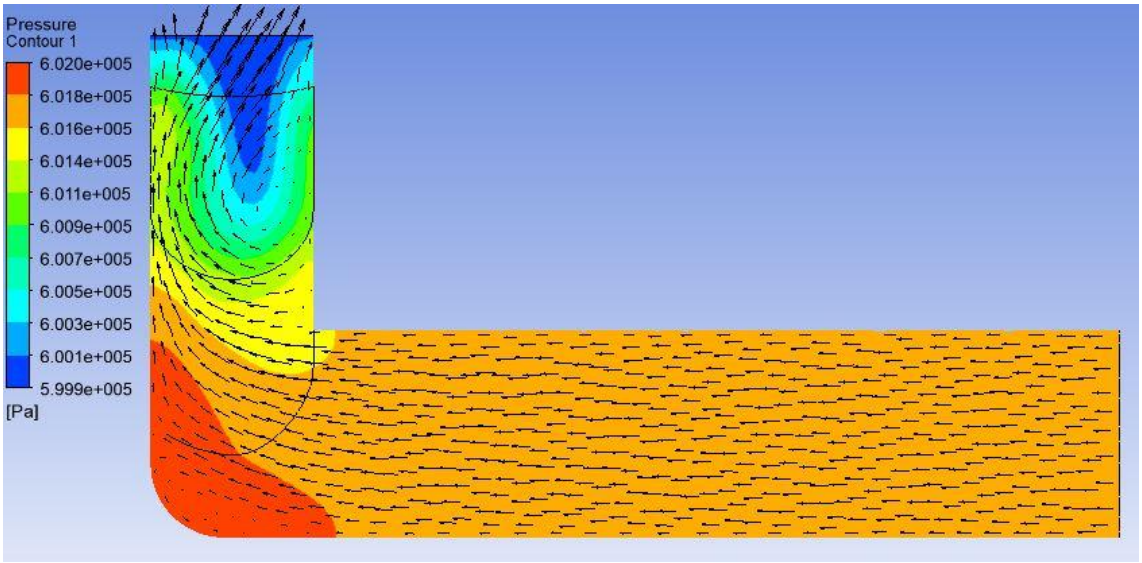


Figure 7.25 – CFD analysis of the flow through the second prototype's inlet shaft (flow pressure contours and velocity vectors)

7.5. Second Wankel Expander Prototype (Mk2): Results

The testing of the second expander went much smoother than the first. This was partly due to experimental techniques already being developed and partly because the second prototype had very little problems during operation. The only significant issue was when some small debris entered the prototype and jammed between the rotor and shaft. This incident caused major scoring of the shaft and rotor interface surfaces, which required extensive sanding and polishing in order to operate again. This is a good motivation to ensure the inlet gas is very well filtered, as even a small particle can cause extensive damage due to the small clearances and high speeds involved.

The following results, like for the first expander, have been grouped into inlet pressures to the nearest 0.1bar to allow comparison.

7.5.1. Friction Test Results

Friction tests were carried out using the same method as the first prototype. Figure 7.26 shows a comparison between the friction test results for the first and second prototypes. It can be easily noticed that there is much greater friction in the second expander. This is thought to be mainly due to the additional bearing causing friction, plus the added weight of the bearings causing additional friction in all the other bearings of the expander. It is also worth mentioning that all of the ball bearings are lubricated with packed grease. If a lower friction lubrication alternative could be used in the bearings, then the friction in both prototypes could be significantly reduced.

Another contributing factor could be the gears. In the first prototype, steel gears bought from an external gear company and modified for the expander were used. However, in the second expander, plastic 3D printed gears were utilised due to manufacturing time constraints. Although there are not large loads on the gears, the surface finish of the plastic gears may be producing additional friction.

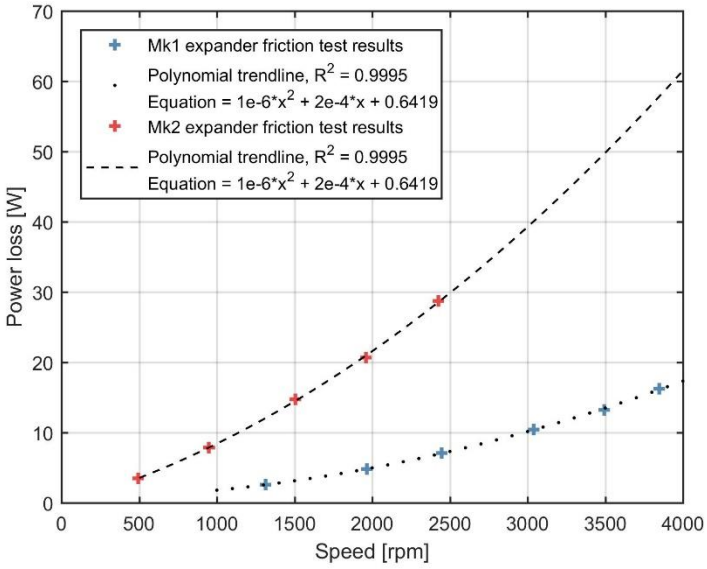


Figure 7.26 – Friction power loss test results for 1st and 2nd prototype

7.5.2. Comparison to First Prototype

Figures 7.27 and 7.28 compare the results from the second prototype with the unmodified and modified first prototypes, respectively. As the results for the first prototype did not have efficiency results, only the power outputs are compared. When compared to the unmodified first prototype, the power output of the second prototype is greater in all cases. The improvement is much greater for larger inlet pressures, at 0.3bar inlet pressure, there is not much difference, but at 0.8bar the power output is more than doubled.

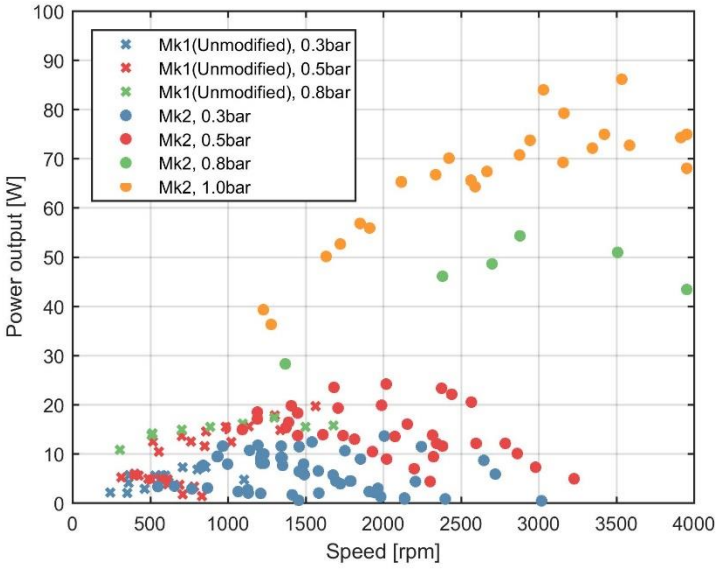


Figure 7.27 – Experimental results comparing the first (unmodified) and second expander prototypes

Comparing to the modified first prototype, the second prototype has much less of a performance benefit for the same inlet pressures. The modified first expander has a larger power output at an inlet pressure of 0.3bar. This is because the friction problems of the first expander did not affect it much at low speeds and the second expander has more measured frictional losses. At 0.5bar the modified first prototype continues to have a larger power output and at 0.6bar both prototypes have very similar power outputs. The major benefit to the second prototype is seen in its ability to withstand larger inlet pressures, as can be seen by the 1bar inlet pressure results, which the first prototype expander was not capable of and would stop rotating from pressure induced friction. During the tests the second prototype ran smoothly at 1bar inlet pressure, and the performance should increase further with larger inlet pressures. Unfortunately, the pressure supply line could only supply a

maximum of between 1-1.2bar inlet gauge pressure at the required flow rate and therefore testing this hypothesis was not possible at the time.

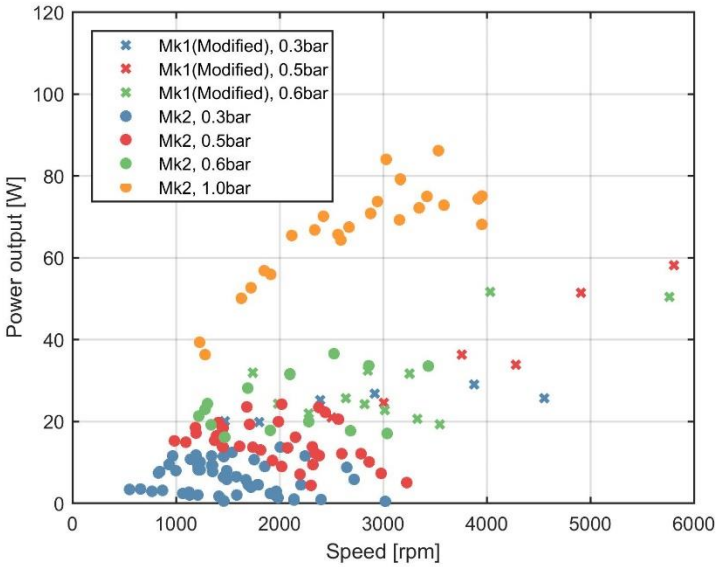


Figure 7.28 – Experimental results comparing the first (modified) and second expander prototypes

7.5.3. Plastic 3D Printed Shaft and Rotor

Before the manufacture of the new rotor and internal shaft parts of the second prototype, they were 3D printed to check for assembly problems and to perform preliminary tests. However, this also presented a good opportunity to compare the performance when the rotor and shaft are manufacture from different materials. The power output results of this comparison are shown in Figure 7.29. Unfortunately, there was not yet accurate flow rate measurement when the plastic parts were tested, so efficiency cannot be compared. The power output for the plastic shaft/rotor are larger for every inlet pressure. This is because the plastic shaft/rotor assembly exerts much less load on the bearings which results in much lower frictional losses. The problem with the plastic parts are their strength, when they were operated

at an inlet pressure larger than 0.7bar the plastic shaft sheared. This occurred with two separate 3D printed plastic shafts, so an inlet pressure above 0.7bar is thought to be impossible for this plastic design, whereas the metal design has no problems with this. However, this shows that if a stronger lightweight material can be used, or if the design can be improved to strengthen the failure area, the performance can be improved.

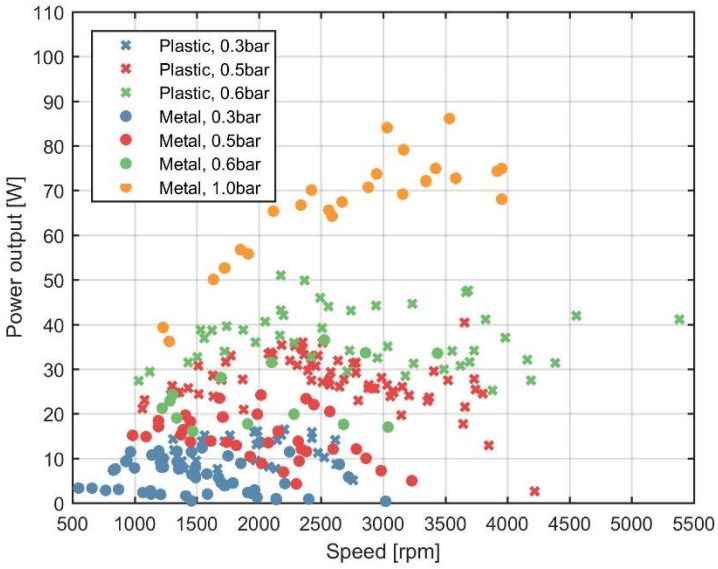


Figure 7.29 – Experimental results comparing the second expander prototype with metal and plastic shaft/rotor assemblies

7.5.4. Comparison to CFD

The following figures compare the experimental results to the results from CFD plus the CFD results modified with frictional losses. Some difference in the results is to be expected as the experimental prototype contained a small clearance between the rotor and housing side faces, whereas the CFD models assumed no leakage at this location. Error bars are included for the power and efficiency experimental results, based on the uncertainty propagation

calculations, they are not presented for the flow rate results, as they were too small to be seen. A 2nd order polynomial trendline is also included for each experimental dataset. Figures 7.30-7.33 show the power output results for 0.5bar, 0.6bar, 0.8bar and 1.0bar inlet gauge pressures. The CFD results adjusted for friction show good agreement with the experimental power results, as they all follow very similar trends. Most experimental power output results lie below the CFD +friction line. It is expected that the experimental results should be less than CFD, as the CFD model includes certain assumptions such as zero wall roughness, zero heat transfer and no leakage past the rotor sides. Furthermore, there will be some unaccounted pressure losses between the inlet sensors and the true inlet to the expander, plus the same between the outlet and outlet sensors. The results lying above could be due to the variations in inlet pressures which were grouped for comparison or the uncertainty of the pressure sensors.

Figures 7.34-7.37 show the efficiency results for 0.5bar, 0.6bar, 0.8bar and 1.0bar inlet gauge pressures. When comparing the trend shape of the experimental results to the CFD with friction losses, it is noticed they are very similar as well. However, in all cases the efficiency predicted by the CFD appears to be almost double that which the experiments measure. This is because the measured experimental flow rates are always approximately double that of the CFD results, as shown in Figures 7.38-7.41. This suggests there is extensive leakage occurring within the expander that is not accounted for by the CFD model. This is most likely the leakage past the rotor

sides, which was not included in the CFD simulations because it would have increased the simulation time excessively. Table 7.2 shows the average deviations of the experimental results from the CFD +friction results for each of the inlet pressures.

Table 7.2 – Average percentage deviations of the experimental results from the CFD results +friction

Inlet pressure	Power output	Isentropic efficiency	Mass flow rate
0.5bar	±24.2% (±3.2W)	-7.7%	+45.9% (+0.0036Kg/s)
0.6bar	±25.3% (±6.1W)	-8.4%	+47.9% (+0.0044Kg/s)
0.8bar	±14.2% (±6.4W)	-11.9%	+48.1% (+0.0055Kg/s)
1.0bar	±8.9% (±5.9W)	-12.3%	+50.6% (+0.0065Kg/s)
Average deviation for all pressures	±18.1% (±5.4W)	-10.1%	+48.1% (+0.0050Kg/s)

Therefore, although the CFD simulations predict the power output of the expander with good accuracy, they would have to be modified to provide more accurate leakage modelling if used with this current design. On the other hand, the experimental prototype could be designed with a smaller clearance gap between the rotor and housing sides or with rotor side seals to reduce this leakage to levels where it becomes insignificant. Additionally, if the expander could be run at a higher speed and inlet pressure, the reduced time for leakage could reduce its contribution to efficiency loss. Finally, if a test rig could be designed that could supply lubrication oil for the expander, this could dramatically reduce leakage, both at the rotor sides and at the rotor apexes, with the bonus that it would reduce friction within the expander.

Power output results

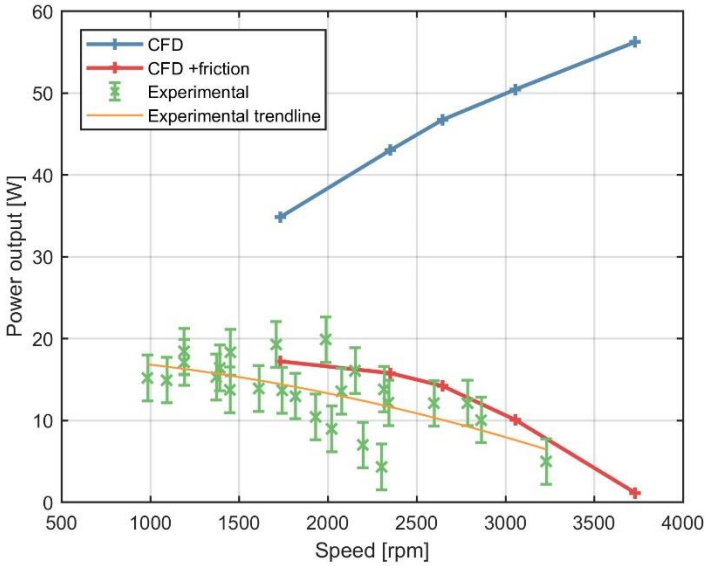


Figure 7.30 - Comparison of CFD and experimental power output results for 0.5bar inlet gauge pressure

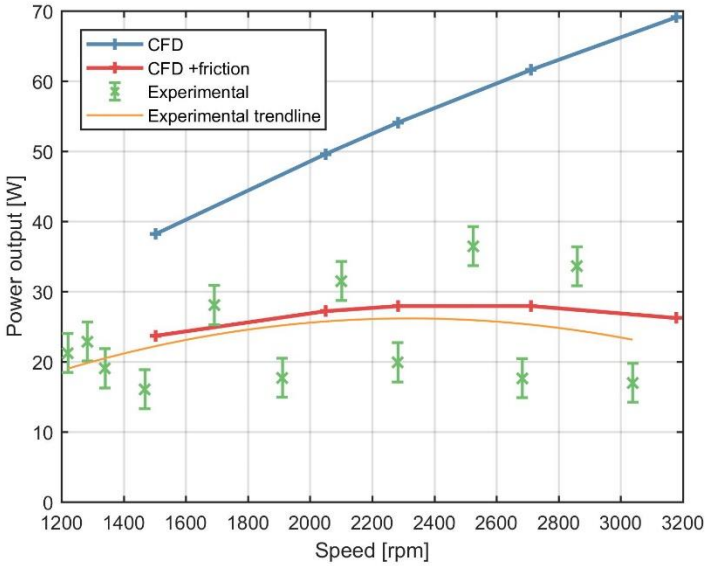


Figure 7.31 - Comparison of CFD and experimental power output results for 0.6bar inlet gauge pressure

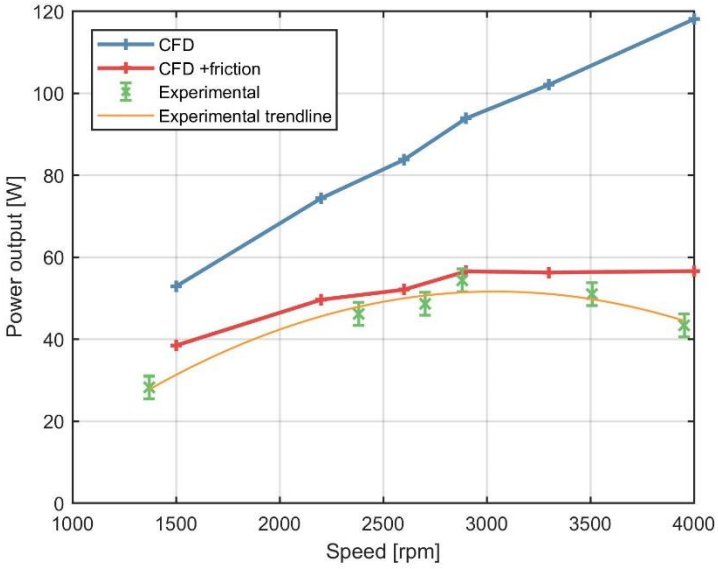


Figure 7.32 - Comparison of CFD and experimental power output results for 0.8bar inlet gauge pressure

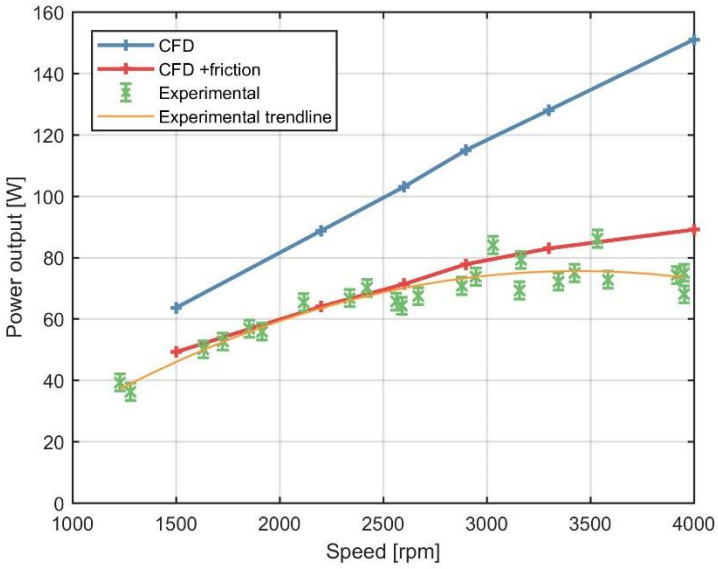


Figure 7.33 - Comparison of CFD and experimental power output results for 1.0bar inlet gauge pressure

Efficiency results

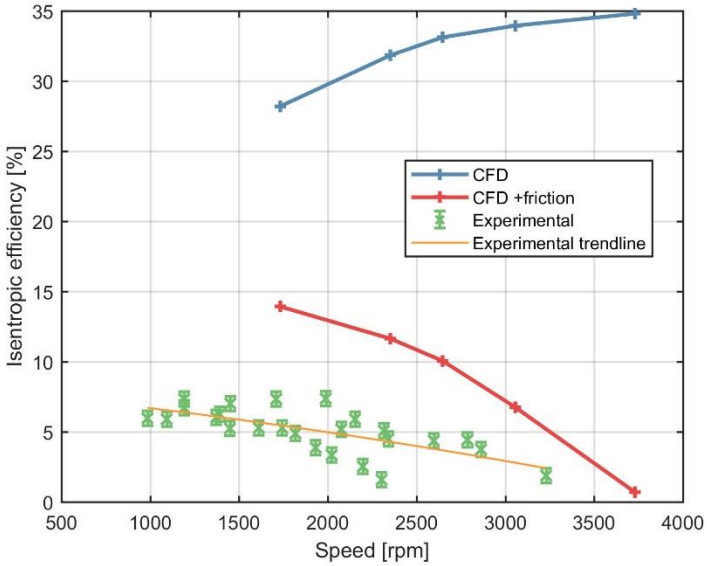


Figure 7.34 – Comparison of CFD and experimental efficiency results for 0.5bar inlet gauge pressure

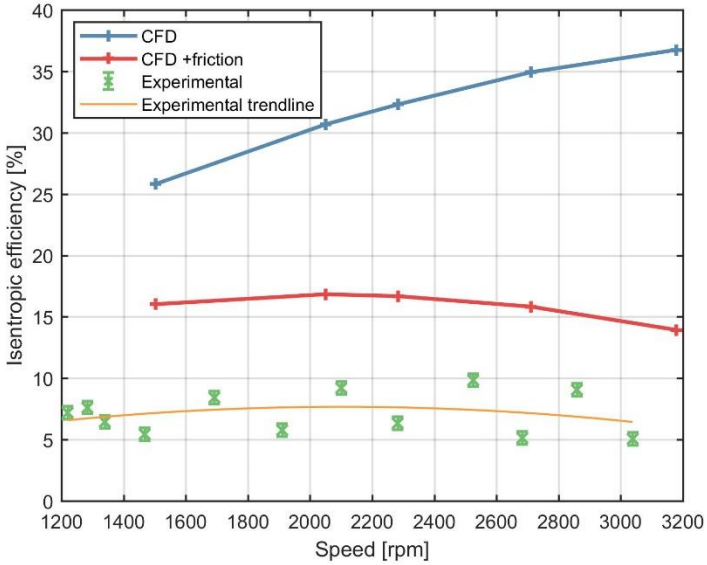


Figure 7.35 - Comparison of CFD and experimental efficiency results for 0.6bar inlet gauge pressure

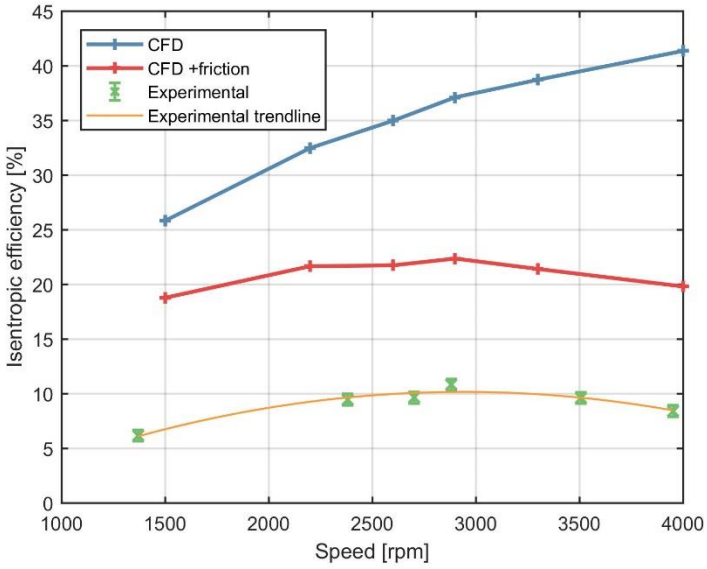


Figure 7.36 - Comparison of CFD and experimental efficiency results for 0.8bar inlet gauge pressure

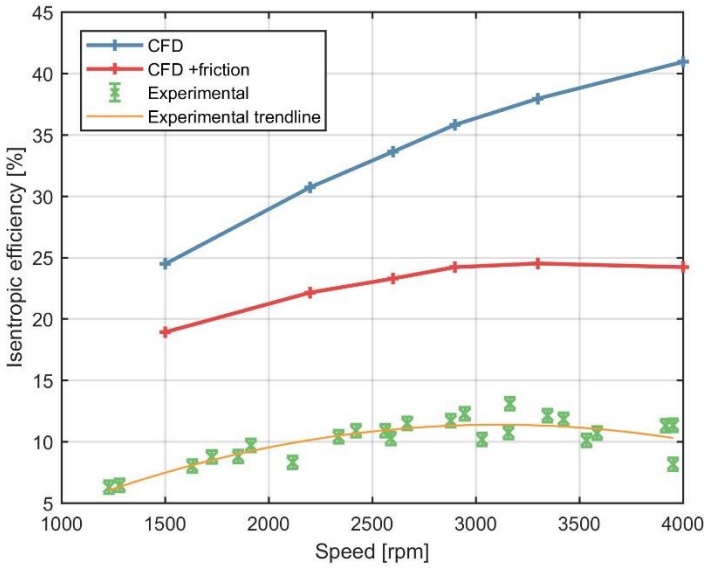


Figure 7.37 - Comparison of CFD and experimental efficiency results for 1.0bar inlet gauge pressure

Flow rate results

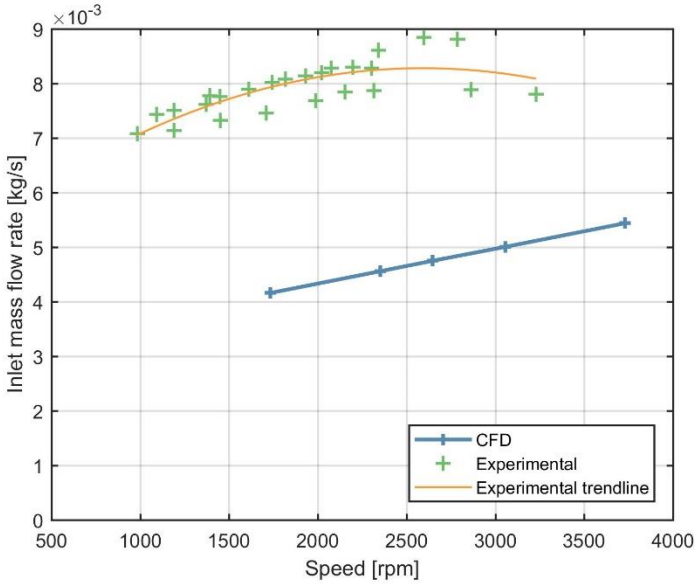


Figure 7.38 - Comparison of CFD and experimental flow rate results for 0.5bar inlet gauge pressure

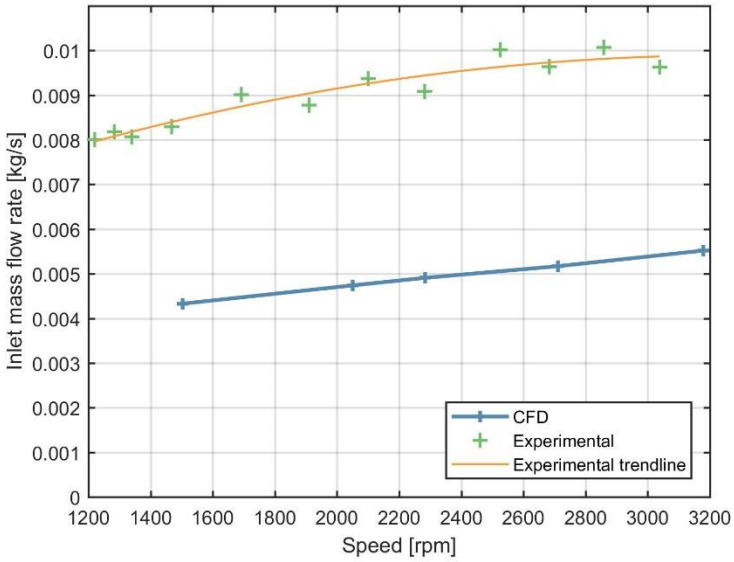


Figure 7.39 - Comparison of CFD and experimental flow rate results for 0.6bar inlet gauge pressure

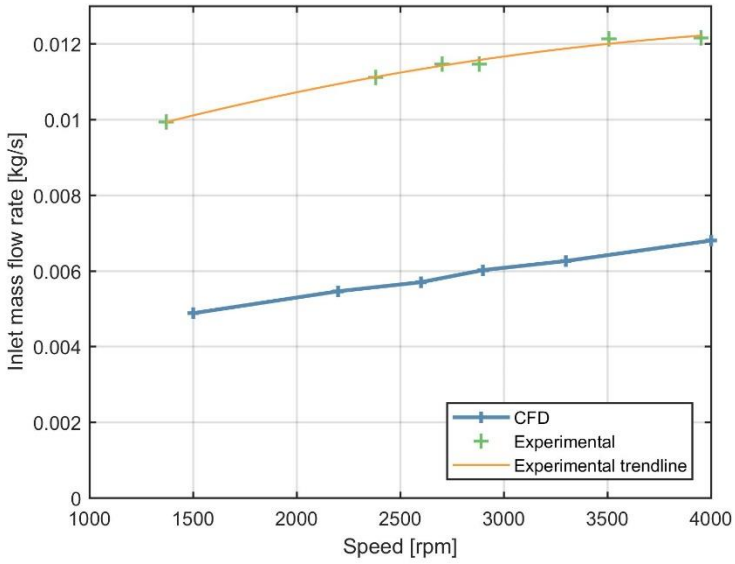


Figure 7.40 - Comparison of CFD and experimental flow rate results for 0.8bar inlet gauge pressure

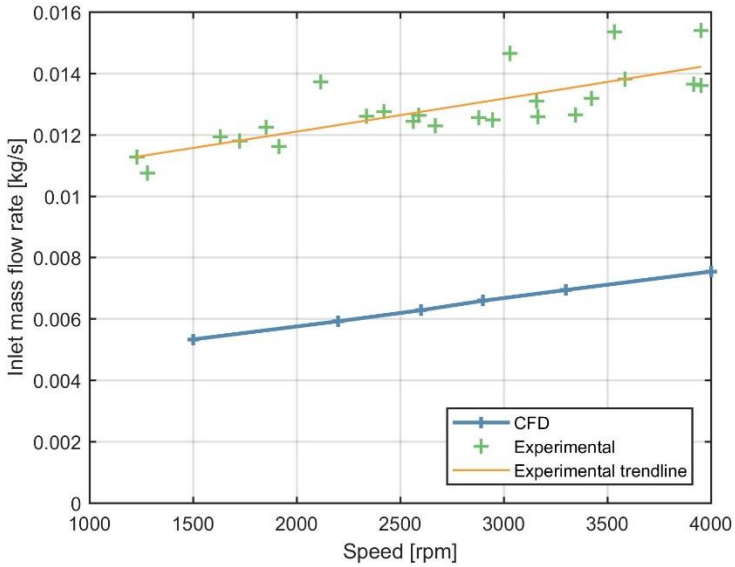


Figure 7.41 - Comparison of CFD and experimental flow rate results for 1.0bar inlet gauge pressure

7.6. Chapter Summary

- The design of both the static shaft Wankel expander prototypes were described, including FEA performed on strength critical parts and CFD on the inlet shaft.

- The first prototype was found to have excess frictional losses when operating at high pressure, the reasons for this friction were described and a modification to the prototype was described. The modification balanced the friction producing forces with other gas forces to reduce the frictional losses. However, losses were still observed at higher pressure.
- A second static shaft Wankel expander prototype was designed to eliminate the source of the large frictional losses. This prototype performed much better and could comfortably operate at high pressures.
- When comparing the results to CFD it was found that the power output had little variation (average of 5.4W) but the efficiency deviated significantly (average of -10.1%). This was thought to be due to unaccounted for leakage in the CFD model (e.g. leakage between the rotor and housing sides).
- The maximum power achieved was 85W and the maximum efficiency achieved was 14%. This was mainly limited by the inlet pressure, as the compressed air supply of the test rig could not consistently reach much higher than 1bar gauge pressure.

CHAPTER 8

CONCLUSIONS

This chapter presents the conclusions reached in the thesis for each chapter's work and this is followed by some recommendations for future work in this area.

8.1. Literature Review Conclusions

Energy

- The worlds energy consumption is following an increasing trend.
- To make best use of renewable energy sources, efficient and cost-effective energy storage systems are required.

LAES

- LAES is a promising new technology for to provide low-cost energy storage.
- Gas compressors and expanders are the most expensive components in LAES. Therefore, development of low-cost, high performance compressors and expanders are required to reach the estimated LAES performances.

Gas Liquefaction

- Small-scale liquefaction could reduce expenditure in the delivery of cryogens. Furthermore, the energy wasted in transporting the cryogen will be removed. Finally, it can utilise onsite renewable energy sources.
- Current small-scale gas liquefaction systems are not very efficient, as they have received little efficiency improvement over the years. Therefore, improving the efficiency of small-scale liquefaction systems is required to make onsite liquefaction for energy storage viable.

ORC

- ORC is a promising up and coming technology providing opportunity to recover wasted thermal energy is many areas.
- Expansion devices are key to the ORC's efficiency. To make small-scale ORC attractive, low-cost, high-efficiency expansion devices should be developed.

Gas expansion devices

- Turbine expanders have highest reported efficiencies but are very costly and have reduced efficiency in small-scale. Screw expanders are more popular for smaller scale where they perform better than turbines, however they are also expensive and hard to manufacture at very small-scale. Turbine and screw expanders have complex 3D geometries which increase manufacturing costs. Scroll expanders are less costly and can compete in performance at small scale, however

they usually require lubricating oil to be mixed in the expansion chamber to reduce friction and leakage.

- Wankel expanders have a simple construction that is easy to manufacture and assemble, they also have low noise and vibrations, a high power to weight ratio and are reliable and durable, which make them ideal for small-scale applications.
- Standard Wankel expanders require added external valves to achieve high efficiency.
- Wankel expander with seals requires lubrication system but have had much development in this field when used as engines.

8.2. Liquefaction Cycles Modelling Conclusions

- Most of the analysed cycles have a very similar peak figure of merit (FOM) values, therefore the Kapitza cycle would be best due to its simplicity.
- At low pressure operation all cycles have an exponentially increasing FOM trend with expander isentropic efficiency, therefore, the efficiency of the expansion device is of prime importance. (the FOM increases with a linear trend for the efficiencies of the compressor and heat exchangers)

8.3. CFD Conclusions

- A new static shaft Wankel expander is described which can remove the compromise between efficiency and simplicity, as an inlet valve can be

incorporated inside the Wankel expander design. Furthermore, it can reduce vibrations further and therefore allow a higher working speed.

- Analysis of the standard Wankel expander gave a maximum isentropic efficiency of 64.88% with 100W power output at 2barg inlet pressure and 1200rpm rotational speed.
- The static shaft Wankel expander gave a maximum isentropic efficiency of 87.3% with 693W power output at 4barg inlet pressure and 8400rpm rotational speed, for the same size expander.
- Performance at different inlet pressures and inlet timings are linked to each other and the static shaft Wankel expander can therefore be easily designed for a large variety of inlet pressures.
- Isentropic efficiency increases almost linearly with rotational speed for cases without apex seals, this is due to the reduced time for leakage. For cases with apex seals, there is a linear decrease in isentropic efficiency with rotational speed (albeit less than the previously mentioned increase) due to the increasing pressure drop over the inlet and outlet ports.
- Reducing apex clearance increases isentropic efficiency in an exponential manner and therefore if apex seals are absent the clearance should be designed as small as possible.
- Using organic working fluids, the maximum efficiency achieved was almost identical to the cases with air as a working fluid, showing the

expander could easily be used in an ORC system without significant modification.

- Out of the two organic working fluids analysed n-Pentane would be preferable for the analysed operating conditions as the expansion occurs closer to its saturation curve.

8.4. Experimental Conclusions

- The first Wankel expander prototype suffered from increased friction due to air pressure pushing moving parts together, the second prototype was designed to remove this problem by adding an additional bearing.
- The power output measurements of the second static shaft Wankel expander prototype matched the CFD results with friction added well (with an average deviation of 5.4W), showing that the previous friction problems had been eliminated.
- For the second static shaft Wankel expander prototype, the efficiency and flow rate showed a relatively large deviation (average deviation of 10.1%) from the CFD results. This a much larger deviation compared with the deviation in power output results for the same prototype. This is thought to be due to internal leakage paths that were not accounted for in the CFD simulations.

8.5. Future Work

- Designing the 2nd prototype static shaft Wankel expander, but with apex seals would be useful to understand if the efficiency increase is as the CFD predicted. A lubrication system would also have to be implemented to properly test this.
- Researching practical methods for manufacturing the static shaft Wankel expander with smaller apex and side clearances to improve the efficiency.
- Optimisation of the materials the static shaft Wankel expander is manufactured from to reduce weight and therefore reduce bearing friction.
- Optimisation of static shaft Wankel expander geometry based on experimental results.
- Run the experimental tests with a higher inlet pressure, to compare to the original higher pressure CFD simulations.
- Modify the CFD model by incorporating side leakage between the rotor and housing sides, this could be through an empirical method, as the CFD may be too computationally heavy.
- Perform a CFD parametric study of the static shaft Wankel expander using organic working fluids without apex seals, to compare to the simulations with air.

- Incorporation of Wankel expander into a gas liquefaction system or ORC system to check the practicalities of using the expander within one of these systems.

BIBLIOGRAPHY

1. International Energy Agency. *Data and statistics (1990-2017).* [accessed 30/04/2020]; Available from: www.iea.org/data-and-statistics.

2. Strahan, D., et al. *Liquid air in the energy and transport systems - opportunities for industry and innovation in the uk*. The centre for low carbon futures partnership; 2013. Full Report. ISBN: 978-0-9575872-1-2.

3. Guizzi, G.L., Manno, M., Tolomei, L.M., and Vitali, R.M., *Thermodynamic analysis of a liquid air energy storage system.* Energy, 2015. **93**: p. 1639-1647.

4. Khalil, K.M., Ahmad, A., Mahmoud, S., and Al-Dadah, R.K., *Liquid air/nitrogen energy storage and power generation system for micro-grid applications.* Journal of Cleaner Production, 2017. **164**: p. 606-617.

5. Rahbar, K., Mahmoud, S., Al-Dadah, R.K., Moazami, N., and Mirhadizadeh, S.A., *Review of organic rankine cycle for small-scale applications.* Energy Conversion and Management, 2017. **134**: p. 135-155.

6. Quoilin, S., Broek, M.V.D., Declaye, S., Dewallef, P., and Lemort, V., *Techno-economic survey of organic rankine cycle (ORC) systems.* Renewable and Sustainable Energy Reviews, 2013. **22**: p. 168-186.

7. Dunn, B., Kamath, H., and Tarascon, J.M., *Electrical energy storage for the grid: A battery of choices.* Science, 2011. **334**(6058): p. 928-35.

8. Rehman, S., Al-Hadhrami, L.M., and Alam, M.M., *Pumped hydro energy storage system: A technological review.* Renewable and Sustainable Energy Reviews, 2015. **44**: p. 586-598.

9. Wu, S., Zhou, C., Doroodchi, E., and Moghtaderi, B., *Techno-economic analysis of an integrated liquid air and thermochemical energy storage system.* Energy Conversion and Management, 2020. **205**.

10. Antonelli, M., Barsali, S., Desideri, U., Giglioli, R., Paganucci, F., and Pasini, G., *Liquid air energy storage: Potential and challenges of hybrid power plants.* Applied Energy, 2017. **194**.

11. Morgan, R., Nelmes, S., Gibson, E., and Brett, G., *Liquid air energy storage – analysis and first results from a pilot scale demonstration plant.* Applied Energy, 2015. **137**: p. 845-853.

12. Morgan, R., Nelmes, S., Gibson, E., and Brett, G., *An analysis of a large-scale liquid air energy storage system.* Proceedings of the Institution of Civil Engineers - Energy, 2015. **168**(2): p. 135-144.

13. Morgan, R.E., *Liquid air energy storage – from theory to demonstration.* International Journal of Environmental Studies, 2016. **73**(3): p. 469-480.

14. Damak, C., Leducq, D., Hoang, H.M., Negro, D., and Delahaye, A., *Liquid air energy storage (LAES) as a large-scale storage technology for renewable energy integration –a review of investigation studies and near perspectives of LAES.* International Journal of Refrigeration, 2020. **110**: p. 208-218.

15. Budt, M., Wolf, D., Span, R., and Yan, J., *A review on compressed air energy storage: Basic principles, past milestones and recent developments.* Applied Energy, 2016. **170**: p. 250-268.

16. Smith, E.M., *Storage of electrical energy using supercritical liquid air.* Proceedings of the lnstitution Mechanical of Engineers, 1977. **191**: p. 289-297.

17. Ordonez, C.A. and Plummer, M.C., *Cold thermal storage and cryogenic heat engines for energy storage applications.* Energy Sources, 1997. **19**(4): p. 389-396.

18. Chino, K. and Araki, H., *Evaluation of energy storage method using liquid air.* Heat Transfer—Asian Research, 2000. **29**(5): p. 347-357.

19. Li, Y., Wang, X., and Ding, Y., *A cryogen-based peak-shaving technology: Systematic approach and techno-economic analysis.* International Journal of Energy Research, 2011. **37**(6): p. 547-557.

20. Ameel, B., T'Joen, C., De Kerpel, K., De Jaeger, P., Huisseune, H., Van Belleghem, M., and De Paepe, M., *Thermodynamic analysis of energy storage with a liquid air rankine cycle.* Applied Thermal Engineering, 2013. **52**(1): p. 130-140.

21. Li, Y., Chen, H., and Ding, Y., *Fundamentals and applications of cryogen as a thermal energy carrier: A critical assessment.* International Journal of Thermal Sciences, 2010. **49**(6): p. 941-949.

22. Evans, A., Strezov, V., and Evans, T.J., *Assessment of utility energy storage options for increased renewable energy penetration.* Renewable and Sustainable Energy Reviews, 2012. **16**(6): p. 4141-4147.

23. Ding, Y., Tong, L., Zhang, P., Li, Y., Radcliffe, J., and Wang, L., *Liquid air energy storage*, in *Storing energy*. 2016. p. 167-181.

24. Legrand, M., Rodríguez-Antón, L.M., Martinez-Arevalo, C., and Gutiérrez-Martín, F., *Integration of liquid air energy storage into the spanish power grid.* Energy, 2019. **187**.

25. Vecchi, A., Li, Y., Mancarella, P., and Sciacovelli, A., *Integrated techno-economic assessment of liquid air energy storage (LAES) under off-design conditions: Links between provision of market services and thermodynamic performance.* Applied Energy, 2020. **262**.

26. Kantharaj, B., Garvey, S., and Pimm, A., *Compressed air energy storage with liquid air capacity extension.* Applied Energy, 2015. **157**: p. 152-164.

27. Li, Y., Wang, X., Jin, Y., and Ding, Y., *An integrated solar-cryogen hybrid power system*. Renewable Energy, 2012. **37**(1): p. 76-81.

28. Zhang, T., Zhang, X.-L., He, Y.-L., Xue, X.-D., and Mei, S.-W., *Thermodynamic analysis of hybrid liquid air energy storage systems based on cascaded storage and effective utilization of compression heat.* Applied Thermal Engineering, 2020. **164**.

29. Georgiou, S., Shahb, N., and Markides, C.N., *A thermo-economic analysis and comparison of pumped-thermal and liquidair electricity storage systems.* Applied Energy, 2018. **226**.

30. Georgiou, S., Aunedi, M., Strbac, G., and Markides, C.N., *On the value of liquid-air and pumped-thermal electricity storage systems in low-carbon electricity systems.* Energy, 2020. **193**.

31. Ordonez, C.A., *Liquid nitrogen fueled, closed brayton cycle cryogenic heat engine.* Energy Conversion & Management, 2000. **41**: p. 331-341.

32. Ahmad, A., Al-Dadah, R., and Mahmoud, S., *Liquid nitrogen energy storage for air conditioning and power generation in domestic applications.* Energy Conversion and Management, 2016. **128**: p. 34-43.

33. Ahmad, A., Al-Dadah, R., and Mahmoud, S., *Air conditioning and power generation for residential applications using liquid nitrogen.* Applied Energy, 2016. **184**: p. 630-640.

34. Ahmad, A., Al-Dadah, R., and Mahmoud, S., *Liquid air utilization in air conditioning and power generating in a commercial building.* Journal of Cleaner Production, 2017. **149**: p. 773-783.

35. Anonymous. *Dearman's liquid air engine could rival battery systems*. Professional Engineering. 2012.

36. Hamdy, S., Morosuk, T., and Tsatsaronis, G., *Exergoeconomic optimization ofan adiabatic cryogenics-based energy storage system.* Energy, 2019. **183**: p. 812-824.

37. Barron, R.F., *Cryogenic systems*. 2nd ed. 1985, Oxford: Oxford University Press.

38. Bisht, V.S., *Thermodynamic analysis of kapitza cycle based on nitrogen liquefaction.* International organization of Scientific Research, 2014. **4**(5): p. 38-44.

39. Abdo, R.F., Pedro, H.T.C., Koury, R.N.N., Machado, L., Coimbra, C.F.M., and Porto, M.P., *Performance evaluation of various cryogenic energy storage systems.* Energy, 2015. **90**: p. 1024-1032.

40. Borri, E., Tafone, A., Romagnoli, A., and Comodi, G., *A preliminary study on the optimal configuration and operating range of a "microgrid scale" air*

liquefaction plant for liquid air energy storage. Energy Conversion and Management, 2017. **143**: p. 275-285.

41. Collins, S.C., *Liquid nitrogen generator.* Review of Scientific Instruments, 1955. **26**(7): p. 671-676.

42. MMR Technologies. *Elan2digital™ liquid nitrogen generators*. [accessed 27/04/2020]; Available from: http://www.elan2.com/products.asp.

43. Noblegen Cryogenics. *Products*. [accessed 27/04/2020]; Available from: http://www.noblegencryogenics.com/.

44. Asynt. *Ln6 liquid nitrogen generator*. [accessed 27/04/2020]; Available from: https://www.asynt.com/product/ln6/.

45. Cryomech. *Liquid nitrogen plants*. [accessed 27/04/2020]; Available from: http://www.cryomech.com/liquid-nitrogen-plants/.

46. Scala. *Liquid nitrogen generator*. [accessed 27/04/2020]; Available from: http://www.scala-filtration.com/products2041112/liquid-nitrogen-generator.htm.

47. Radebaugh, R., *Cryocoolers: The state of the art and recent developments.* J Phys Condens Matter, 2009. **21**(16): p. 164219.

48. Hwang, J. and Lee, K.-Y., *Optimal liquefaction process cycle considering simplicity and efficiency for LNG FPSO at feed stage.* Computers & Chemical Engineering, 2014. **63**: p. 1-33.

49. Venkatarathnam, G., *Cryogenic mixed refrigerant processes*. 2008, New York: Springer. 221-249.

50. Venkatarathnam, G. *Liquefaction of nitrogen using mixed refrigerant processes*. 2008 [accessed 10/09/2020]; Available from: https://www.researchgate.net/publication/267969240_Liquefaction_of_nitrogen_using_mixed_refrigerant_processes.

51. Wang, H.C., Guo, H., Chen, G.F., Zhao, Y.X., Dong, X.Q., Gong, M.Q., and Gao, B., *Thermodynamic design and analysis of movable small scale mixed-refrigerant liquid nitrogen generators.* International Journal of Refrigeration, 2018. **90**: p. 202-215.

52. Nguyen, T.-V., Rothuizen, E.D., Markussen, W.B., and Elmegaard, B., *Thermodynamic comparison of three small-scale gas liquefaction systems.* Applied Thermal Engineering, 2018. **128**: p. 712-724.

53. Badr, O., Naik, S., O'Callaghan, P.W., and Probert, S.D., *Expansion machine for a low power-output steam rankine-cycle engine.* Applied Energy, 1991. **39**(2): p. 93-116.

54. Imran, M., Usman, M., Park, B.-S., and Lee, D.-H., *Volumetric expanders for low grade heat and waste heat recovery applications.* Renewable and Sustainable Energy Reviews, 2016. **57**: p. 1090-1109.

55. Francesconi, M., Dori, E., and Antonelli, M., *Analysis of Balje diagrams for a Wankel expander prototype.* Applied Energy, 2019. **238**: p. 775-785.

56. Zywica, G., Kaczmarczyk, T.Z., and Ihnatowicz, E., *A review of expanders for power generation in small-scale organic rankine cycle systems: Performance and operational aspects.* Proceedings of the Institution of Mechanical Engineers, Part A: Journal of Power and Energy, 2016. **230**(7): p. 669-684.

57. Dumont, O., Dickes, R., and Lemort, V., *Experimental investigation of four volumetric expanders.* Energy Procedia, 2017. **129**: p. 859-866.

58. Vittorini, D., Bianchi, G., and Cipollone, R., *Energy saving potential in existing volumetric rotary compressors.* Energy Procedia, 2015. **81**: p. 1121-1130.

59. Lemort, V. and Legros, A., *Positive displacement expanders for organic rankine cycle systems*, in *Organic rankine cycle (ORC) power systems*. 2017. p. 361-396.

60. Weiß, A.P., *Volumetric expander versus turbine – which is the better choice for small orc plants?,* 3rd International Seminar on ORC Power Systems, Brussels, 2015.22.

61. Qiu, G., Liu, H., and Riffat, S., *Expanders for micro-CHP systems with organic rankine cycle.* Applied Thermal Engineering, 2011. **31**(16): p. 3301-3307.

62. Saghlatoun, S., Zhuge, W., and Zhang, Y., *Review of expander selection for small-scale organic rankine cycle,* ASME 2014 4th Joint US-European Fluids Engineering Division Summer Meeting, Chicago, Illinois, USA, 2014

63. Lemort, V., Guillaume, L., Legros, A., Declaye, S., and Quoilin, S., *A comparison of piston, screw and scroll expanders for smallscale rankine cycle systems,* 3rd International Conference on Microgeneration and Related Technologies, 2013

64. Xia, G.-D., Zhang, Y.-Q., Wu, Y.-T., Ma, C.-F., Ji, W.-N., Liu, S.-W., and Guo, H., *Experimental study on the performance of single-screw expander with different inlet vapor dryness.* Applied Thermal Engineering, 2015. **87**: p. 34-40.

65. Ziviani, D., Suman, A., Lecompte, S., De Paepe, M., van den Broek, M., Spina, P.R., Pinelli, M., Venturini, M., and Beyene, A., *Comparison of a single-screw and a scroll expander under part-load conditions for low-grade heat recovery ORC systems.* Energy Procedia, 2014. **61**: p. 117-120.

66. Wankel, F., *Rotary piston machines*. 1965, London: ILIFFE BOOKS LTD.

67. Jones, C., *A survey of curtiss-wright's 1958-1971 rotating combustion engine technological developments,* National Automobile Engineering Meeting, Detroit, Michigan, 1972

68. Weston, K., *Energy conversion*. 1992, Minnesota, United States: West Publishing Company.

69. Badr, O., Naik, S., O'Callaghan, P.W., and Probert, S.D., *Rotary Wankel engines as expansion devices in steam rankine-cycle engines.* Applied Energy, 1991. **39**(1): p. 59-76.

70. Ogura, I., *The Ogura-Wankel compressor - application of Wankel rotary concept as automotive air conditioning compressor,* International Congress & Exposition, Detroit, 1982

71. Omirou, S.L. and Nearchou, A.C., *An epitrochoidal pocket—a new canned cycle for CNC milling machines.* Robotics and Computer-Integrated Manufacturing, 2009. **25**(1): p. 73-80.

72. Beard, J. and Pennock, G., *The effects of design parameters on the displacement and compression ratio of the Wankel rotary compressor. ,* International Compressor Engineering Conference, Purdue Univeristy, Indiana, 1990.686.

73. Beard, J.E. and Pennock, G.R., *Calculation of the displacernent of a Wankel rotary compressor,* International Compressor Engineering Conference, Purdue Univeristy, Indiana, 1992.788.

74. Weerasinghe, R. and Hounsham, S., *Small engines as bottoming cycle steam expanders for internal combustion engines.* Journal of Combustion, 2017. **2017**: p. 1-8.

75. Wu, Y., Ma, C., Chen, X., and Du, C., *Optimized design for micro Wankel compressor used in space-borne vapor compression heat pump,* International Compressor Engineering Conference, Purdue Univeristy, Indiana, 2014.2304.

76. Antonelli, M. and Martorano, L., *A study on the rotary steam engine for distributed generation in small size power plants.* Applied Energy, 2012. **97**: p. 642-647.

77. Antonelli, M., Baccioli, A., Francesconi, M., Desideri, U., and Martorano, L., *Operating maps of a rotary engine used as an expander for micro-generation with various working fluids.* Applied Energy, 2014. **113**: p. 742-750.

78. Xu, X., Xu, H., Deng, H., Gu, F., and Talbot, C., *An investigation of a hypocycloid mechanism based twin-rotor piston engine.* Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2014. **229**(1): p. 106-115.

79. Spreitzer, J., Zahradnik, F., and Geringer, B., *Implementation of a rotary engine (Wankel engine) in a CFD simulation tool with special emphasis on combustion and flow phenomena,* SAE Technical Paper Series, Detroit, 2015

80. Becker, E. *What stalled the Wankel*. Tribology and Lubrication Technology [Online] 2016 [accessed 11/11/2020]; Available from: http://digitaleditions.walsworthprintgroup.com/publication/?i=319907&article_id=2531236&view=articleBrowser&ver=html5.

81. Fu, K., Knobloch, A., Martinez, F.C., Walther, D., Fernandez-pello, A., Pisano, A., Liepmann, D., Miyaska, K., and Maruta, K., *Design and experimental results of small-scale rotary engine,* ASME International Mechanical Engineering Congress and Exposition, New York, 2001

82. Antonelli, M., Baccioli, A., Francesconi, M., and Martorano, L., *Experimental and numerical analysis of the valve timing effects on the performances of a small volumetric rotary expansion device,* 68th Conference of the Italian Thermal Machines Engineering Association, Bologna, 2014

83. Zhong, X., Gou, Y., Wu, Y., and Ma, C., *Development and experimental study of a miniature vapor compression refrigeration equipment.* Science in China Series E: Technological Sciences, 2008. **51**(5): p. 632-640.

84. Wu, Y., Ma, C., and Zhong, X., *Development and experimental investigation of a miniature-scale refrigeration system.* Energy Conversion and Management, 2010. **51**(1): p. 81-88.

85. Badr, O., Naik, S., O'Callaghan, P.W., and Probert, S.D., *Wankel engines as steam expanders: Design considerations.* Applied Energy, 1991. **40**(3): p. 157-170.

86. Antonelli, M., Baccioli, A., Francesconi, M., and Martorano, L., *Numerical and experimental analysis of the intake and exhaust valves of a rotary expansion device for micro generation.* Energy Procedia, 2015. **81**: p. 461-471.

87. Antonelli, M., Francesconi, M., Baccioli, A., and Caposciutti, G., *Experimental results of a Wankel-type expander fuelled by compressed air and saturated steam.* Energy Procedia, 2017. **105**: p. 2929-2934.

88. Zhang, Y.L. and Wang, W., *Efficiency enhancement in a meso-Wankel compressor with overflow.* Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science, 2011. **226**(6): p. 1550-1567.

89. Francesconi, M. and Antonelli, M., *A numerical model for the prediction of the fluid dynamic and mechanical losses of a Wankel-type expansion device.* Applied Energy, 2017. **205**: p. 225-235.

90. Zhang, Y. and Wang, W., *Effects of leakage and friction on the miniaturization of a Wankel compressor.* Frontiers in Energy, 2010. **5**(1): p. 83-92.

91. Zhang, Y. and Wang, W., *Friction loss and wear in a meso scale Wankel compressor.* Journal of Fluids Engineering, 2013. **135**(8).

92. Rose, S.W. and Yang, D.C.H., *Wide and multiple apex seals for the rotary engine.* Mechanism and Machine Theory, 2014. **74**: p. 202-215.

93. Pennock, G.R. and Beard, J.E., *Force analysis of the apex seals in the Wankel rotary compressor including the influence of fluctuations in the crankshaft speed.* Mechanism and Machine Theory, 1997. **32**(3): p. 349-361.

94. Zhang, D.-l., Wu, Y.-t., Wang, J.-f., Du, C.-x., Chen, X., Ma, R., and Ma, C.-f., *Theoretical study of seal spring in a Wankel compressor,* 23rd International Compressor Engineering Conference Purdue, 2016

95. Picard, M., Tian, T., and Nishino, T., *Predicting gas leakage in the rotary engine—part ii: Side seals and summary.* Journal of Engineering for Gas Turbines and Power, 2016. **138**(6).

96. Phung, T.H., Sultan, I.A., and Appuhamillage, G.K., *On the apex seal analysis of limaçon positive displacement machines.* Mechanism and Machine Theory, 2018. **127**: p. 126-145.

97. Zhang, S., Liu, J., and Zhou, Y., *Effect of DLC coating on the friction power loss between apex seal and housing in small Wankel rotary engine.* Tribology International, 2019. **134**: p. 365-371.

98. Zhang, S., Liu, J., Zuo, Z., and Zhang, Y., *An analytical investigation of oil film thickness for the apex seal in a small Wankel rotary engine.* Tribology International, 2017. **116**: p. 383-393.

99. Robertson, G.F. and Woigemuth, C.H. *Analysis and test apparatus for a vane expander using steam*. in *Proc. 10th IECEC*. 1975. Newark, USA.

100. Heppner, J.D., Walther, D.C., and Pisano, A.P., *The design of arctic: A rotary compressor thermally insulated μcooler.* Sensors and Actuators A: Physical, 2007. **134**(1): p. 47-56.

101. Garside, D.W., *A new Wankel-type compressor and vacuum pump.* IOP Conference Series: Materials Science and Engineering, 2017. **232**.

102. Wappenschmidt, J., Sonntag, S.J., Buesen, M., Gross-Hardt, S., Kaufmann, T., Schmitz-Rode, T., Autschbach, R., and Goetzenich, A., *Fluid dynamics in rotary piston blood pumps.* Ann Biomed Eng, 2017. **45**(3): p. 554-566.

103. Wan, S., Lau, M.W.S., and Goh, K.L., *Gearless Wankel-like pump/mixer: Design challenges and prospects,* 3rd International Conference on Power Generation Systems and Renewable Energy Technologies (PGSRET), Johor Bahru, 2017

104. Stead, I.M.N., Roberts, A., Eckold, D.G., and Dearn, K.D., *Cold, clean and green: Improving the efficiency and environmental impact of a cryogenic*

expander. IOP Conference Series: Materials Science and Engineering, 2019. **502**.

105. Zhang, Y. and Wang, W., *Factors affecting the cylinder displacement of a Wankel compressor in a micro-cooling system*. HVAC&R Research, 2012(18:3): p. 366-376.

106. Francesconi, M., Caposciutti, G., and Antonelli, M., *An experimental and numerical analysis of the performances of a Wankel steam expander*. Energy Procedia, 2017. **129**: p. 395-402.

107. Matsuura, K. and Terasaki, K., *Measurement of the relative behavior of a rotary engine apex seal to the side of a slot using magnetic induction*. JSME international journal. Ser. 2, Fluids engineering, heat transfer, power, combustion, thermophysical properties, 1991. **34**(2): p. 264-269.

108. Matsuura, K. and Terasaki, K., *Measurement and theoretical analysis of the underseal pressure of rotary engine apex seals*. JSME international journal. Ser. 2, Fluids engineering, heat transfer, power, combustion, thermophysical properties, 1992. **35**(3): p. 448-457.

109. ANSYS Inc., *ANSYS Fluent user's guide, release 18.2*. 2017.

110. Pico Technology. *Improving the accuracy of temperature measurements*. 2020 [accessed 14/05/2020]; Available from: https://www.picotech.com/library/application-note/improving-the-accuracy-of-temperature-measurements.

111. Kallhovd, O. *Library for the hx711 24-bit adc for weight scales*. 2020 [accessed 18/05/2020]; Available from: https://www.arduinolibraries.info/libraries/hx711_adc.

112. Dejan. *Arduino brushless motor control tutorial | esc | bldc*. 2019 [accessed; Available from: https://howtomechatronics.com/tutorials/arduino/arduino-brushless-motor-control-tutorial-esc-bldc/.

Appendix A: MATLAB Code -

Liquefactions Cycles Models

```
function hpos = StateCheck(hpos,houtlet,pressure,Toutlet,Tpos,flips,gas)
%Checks the state conditions of current and next possible enthalpies to
%prevent errors, outputs the enthalpy
%Requires SPECIFIC ENTHALPY inputs and outputs the same.
%Uses number of oscillations (flips) to reduce gradient and help convergence

if pressure >= CoolProp.Props1SI('Pcrit',gas)
    return;
end

hSatGas = CoolProp.PropsSI('H','P',pressure,'Q',1,gas);
hSatLiq = CoolProp.PropsSI('H','P',pressure,'Q',0,gas);
TBoiling = CoolProp.PropsSI('T','P',pressure,'Q',1,gas);
x = (hSatGas - hSatLiq) / (20 * flips);
%If pressure below critical, Hpossible is at boiling temp decrease or increase
slowly over range
%also if Hpos is liquid and Hcurrent is gas or visa versa
if (houtlet > hSatGas && hpos < hSatLiq) || (houtlet < hSatLiq && hpos > hSatGas)
|| abs(TBoiling - Tpos) < 0.001 || abs(TBoiling - Toutlet) < 0.001

    if hpos < houtlet
        hpos = houtlet - x;
    else
        hpos = houtlet + x;
    end
end
```

```
function [H] = VapCheck(T,P,gas)
%Calculates enthalpy based on pressure and temperature
if T < CoolProp.Props1SI('Tcrit',gas)

    if          P<CoolProp.Props1SI('Pcrit',gas)          &&          T          <
CoolProp.PropsSI('T','P',P,'Q',1,gas)+        0.001        &&        T        >
CoolProp.PropsSI('T','P',P,'Q',0,gas) -0.001
        %if p and t values within mixture range, possible h set as coolest (q=0)
        H  =  CoolProp.PropsSI('H','P',P,'Q',0,gas);%Enthalpy  equals  that  of
saturated

    else
        H = CoolProp.PropsSI('H','P',P,'T',T,gas); %if below critial temperature
and pressure and T is not at boiling
    end
else
    H = CoolProp.PropsSI('H','P',P,'T',T,gas);
End
```

```
function [Hnew] = UpdateGrad(prev,cur,new)
%2.changes the new enthalpy to be closer to the previous to help
%convergence
dH = cur.H - prev.H; %previous change in enthalpy
```

```
dHpos = new.HP - cur.H; %new change in enthalpy, based on calculated enthalpy
dHnew = (dH +dHpos)/2; %take the mean of the previous and new changes in enthalpy
Hnew = cur.H + dHnew; %Update the new enthalpy based on the calculated change
```

```
function L2 = HPUpdate2(L1,gas)
%Updates the state point based on the enthalpy and pressure values
L1.T = CoolProp.PropsSI('T','P',L1.P,'H',L1.H,gas);
L1.S = CoolProp.PropsSI('S','P',L1.P,'H',L1.H,gas);
L1.Q = CoolProp.PropsSI('Q','P',L1.P,'H',L1.H,gas);
if L1.Q == -1 %if state is not in vapour/liquid mixture state
    if L1.P < (CoolProp.Props1SI('Pcrit',gas))
        if L1.H > CoolProp.PropsSI('H','P',L1.P,'Q',1,gas)
            L1.Q = 1; %if not critical and saturated gas
        end

        if L1.H < CoolProp.PropsSI('H','P',L1.P,'Q',0,gas)
            L1.Q = 0; %if not critical and saturated liquid
        end
    else
        L1.Q = 1; %if above critical pressure
    end
end

L2.T = L1.T;
L2.P = L1.P;
L2.H = L1.H;
L2.S = L1.S;
L2.Q = L1.Q;
L2.F = L1.F;
L2.HP = L1.HP;
L2.R = L1.R;
L2.RF = L1.RF;
```

```
function [Hout,Wout] = Expander(Lin,Lout,eff,gas)
%Expander function
Hin = Lin.H;
Houtisen = CoolProp.PropsSI('H','S',Lin.S,'P',Lout.P,gas);
Hout = Hin- (eff*(Hin-Houtisen));
Wout = Lin.F * (Hin - Hout); %*Power output
```

```
function [Work] = IdealLiqueWork2(P,T,gas)
%Finds the ideal work to liquefy the gas (for efficiency calc)
%P and T are the (ambient) pressure and temperature

Sf = CoolProp.PropsSI('S','Q',0,'P',P,gas);
Hf = CoolProp.PropsSI('H','Q',0,'P',P,gas);

S1 = CoolProp.PropsSI('S','T',T,'P',P,gas);
H1 = CoolProp.PropsSI('H','T',T,'P',P,gas);

Work = (T*(S1-Sf)) - (H1-Hf);
```

```
function L2 = InitH2(L1,P,T,gas)
%Initialises a state point based on pressure and temperature

L1.P = P;
L1.T = T;
L1.H = CoolProp.PropsSI('H','T',T,'P',P,gas);
L1.S = CoolProp.PropsSI('S','T',T,'P',P,gas);
L1.Q = CoolProp.PropsSI('Q','T',T,'P',P,gas);
```

```
L1.R = 10000;
L1.RF =10000;

if L1.Q == -1
    L1.Q = 1;
end

L2.H = L1.H;
L2.T = L1.T;
L2.P = L1.P;
L2.S = L1.S;
L2.Q = L1.Q;
L2.F = L1.F;
L2.HP = L1.H;
L2.R = L1.R;
L2.RF = L1.RF;
```

```
function [Htotal] = MixedStreams3(L1,L2)
%Calculates the resultant enthalpy when two stream are joined into one
Htotal = ((L1.F*L1.H)+(L2.F*L2.H))/(L1.F+L2.F);% outlet enthalpy per kg
```

```
function                                [h1out,h2out]                                =
HeatExchanger11(inlet1,inlet2,outlet1,outlet2,E,flips,gas)
%Heat exchanger function

if outlet1.F == 0 || outlet2.F == 0 %if no flow condition
    h1out = inlet1.H;
    h2out = inlet2.H;
    return
end

H1in = inlet1.H * outlet1.F; %Total enthalpy at inlet1
h1outpos  =  VapCheck(inlet2.T,outlet1.P,gas);  %Possible  SPECIFIC  enthalpy  at
outlet1 if cooled to inlet2

%If possible is at boiling temp decrease or increase slowly over range
%also if Hpos is liquid and Hcurrent is gas or visa versa
H1outpos                                                                        =
StateCheck(h1outpos,outlet1.H,outlet1.P,outlet1.T,inlet2.T,flips,gas)           *
outlet1.F;

H2in = inlet2.H * outlet2.F; %Total enthalpy at inlet2
h2outpos  =  VapCheck(inlet1.T,outlet2.P,gas);  %Possible  SPECIFIC  enthalpy  at
outlet2 if cooled to inlet1

%If possible is at boiling temp decrease or increase slowly over range
H2outpos                                                                        =
StateCheck(h2outpos,outlet2.H,outlet2.P,outlet2.T,inlet1.T,flips,gas)           *
outlet2.F;

%Updating enthalpies
dH1 = H1in-H1outpos; %Possible total enthalpy change 1
dH2 = H2in-H2outpos; %Possible total enthalpy change 2

%Choose lowest total enthalpy change and multiply by effectiveness
if abs(dH1) <= abs(dH2)
    dH1 = E * dH1;
    dH2 = -dH1; %*-1 as opposite direction heat exchange
else
    dH2 = E * dH2;
    dH1 = -dH2; %*-1 as opposite direction heat exchange
```

```
end

%Set output specific enthalpys
h1out = (H1in -dH1) / outlet1.F;
h2out = (H2in -dH2) / outlet2.F;
```

```
function [yield,work,FOM,maxr] = SSClaude5(input,func)
%Claude.

%func=...     0 for not function || 'P' for pressure || 'x' for x || 'E' for
exchanger eff ||
%    'd' for expander eff || 'c' for compressor eff

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0 %if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
gas = 'N2';
NoLoc = 14; %number of locations
HiLoc = 6; %number of high pressure locations
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E =1; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.75; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 6000; %Maximum iterations
flips = 30;%Number of oscillations before force end
mul = 0.1;  %Convergence factor multiplier

%Convergence multipliers
con = zeros(NoLoc,1);
conf = zeros(NoLoc,1);
for i=1:NoLoc
    con(i) = 1.5;
    conf(i) = 1.5;
end

%Analysis function switches
switch func
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
```

```
    case 0
    otherwise
        error("Incorrect input selector\n");
end


%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Main Data structure. Syntax -- l(location,iteration).parameter
l = struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[]);
lfpos = zeros(NoLoc);

%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
%3 -- before split, after 1st exchanger (hot stream)
%4 -- before expander
%5 -- after split, before 2nd exchanger (hot stream)
%6 -- after 2nd exchanger, before 3rd exchanger (hot stream)
%7 -- after 3rd exchanger (hot stream), before valve
%8 -- after valve, before separator
%9 -- after separator (gas), before 3rd exchanger (cold stream)
%10 -- after 3rd exchanger (cold stream), before join
%11 -- after expander
%12 -- after join, before 2nd exchanger (cold stream)
%13 -- after 2nd exchanger (cold stream), before 1st exchanger (cold
%stream)
%14 after 1st exchanger (cold stream)

%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e split
f2 = f1*(1-x);
%e flow
fe = f1 * x;

for i=1:NoLoc
    l(i,1).F = fe;%initialise all as e, as they are spread
end
for i=1:3
    l(i,1).F = f1; %then correct others in groups
end
for i=5:10
```

```
    l(i,1).F = f2;
end
for i=12:14
    l(i,1).F = f1;
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end

%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H); %per kg
isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork (work per kg)

c=2;
converged =0;
NoFlips = 1;
breaking = 0;
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %3rd heat exchanger 6 7 9 10

    [l(7,c+1).HP,l(10,c+1).HP]                                                    =
HeatExchanger11(l(6,c),l(9,c),l(7,c),l(10,c),E,NoFlips,gas);
    lfpos(7) = l(6,c).F;
    lfpos(10) = l(9,c).F;

    %2nd heat exchanger 5 6 12 13

    [l(6,c+1).HP,l(13,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(12,c),l(6,c),l(13,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(13) = l(12,c).F;

    %1st heat exchanger 2 3 13 14
```

```
    [l(3,c+1).HP,l(14,c+1).HP]                                                    =
HeatExchanger11(l(2,1),l(13,c),l(3,c),l(14,c),E,NoFlips,gas);
    lfpos(3) = l(2,1).F;
    lfpos(14) = l(13,c).F;

    %Expander 4->11

    [l(11,c+1).HP,workExp] = Expander(l(4,c),l(11,c),ExpEff,gas);
    lfpos(11) = l(4,c).F;

    %Mixing streams 8 and e -> 9  10&11 = 12

    l(12,c+1).HP = MixedStreams3(l(10,c), l(11,c));
    lfpos(12) = l(10,c).F + l(11,c).F;

    %Split streams

    l(5,c+1).HP = l(3,c).H;
    lfpos(5) = l(3,c).F * (1-x);

    l(4,c+1).HP = l(3,c).H;
    lfpos(4) = l(3,c).F * x;


    %JT expansion 7-8

    l(8,c+1).HP = l(7,c).H;
    lfpos(8) = l(7,c).F;

    %Seperator 6->7 8-9

    if l(8,c).Q == 1
        l(9,c+1).HP = l(8,c).H;
    else
        l(9,c+1).HP = CoolProp.PropsSI('H','P',l(9,c).P,'Q',1,gas);
    end
    lfpos(9) = l(8,c).F * l(8,c).Q;

    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end


    %enthalpy residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for each
location
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for enth
residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E ==1
                NoFlips = NoFlips + 1;
            end
        end
```

```
        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for flip
            conf(i) = conf(i) + mul;
        end
    end

    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);



            l(8,c).Q = mean([l(8,round(2*c/3):c).Q]); %After valve
            l(11,c).H = mean([l(11,round(2*c/3):c).H]); %After Expander
            l(4,c).F = mean([l(4,round(2*c/3):c).F]); %Before expander
            l(4,c).H = mean([l(4,round(2*c/3):c).H]);
            [l(11,c+1).HP,workExp]    =    Expander(l(4,c),l(11,c),ExpEff,gas);
%before and after expander
            breaking=1;
            break
        end
    end

    if breaking ==1
        break %also break out of while
    end

    %Covergence checks
    if c>NoLoc %After number of locations worth of iterations, start checking
for convergence
        converged = 1; %Set to yes
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                maxr = 1;
                converged = 0; %Change back if any aren't converged
            end
        end
    end
    c = c+1;
end

yield = (1-x)*(1-l(8,c).Q);

work =  ((workComp) - (workExp)) / yield; %%

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-[l(8,:).Q]));
    xlabel('Iteration')
```

```
    ylabel('Yield')


    figure
    plot([l(3,:).H],'DisplayName','H3');
    hold on
    plot([l(4,:).H],'DisplayName','H4');
    plot([l(5,:).H],'DisplayName','H5');
    plot([l(6,:).H],'DisplayName','H6');
    plot([l(7,:).H],'DisplayName','H7');
    if l(2,c).P < CoolProp.Props1SI('Pcrit',gas)

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',0,gas)]);
    end
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off


    figure
    plot([l(8,:).H],'DisplayName','H8');
    hold on
    plot([l(9,:).H],'DisplayName','H9');
    plot([l(10,:).H],'DisplayName','H10');
    plot([l(11,:).H],'DisplayName','H11');
    plot([l(12,:).H],'DisplayName','H12');
    plot([l(13,:).H],'DisplayName','H13');
    plot([l(14,:).H],'DisplayName','H14');

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',0,gas)]);
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

```
function [yield,work,FOM,maxr] = SSKapitza5(input,func)
%Kapitza cycle.

%func=...     0 for not function || 'P' for pressure || 'x' for x || 'E' for
exchanger eff ||
%    'd' for expander eff || 'c' for compressor eff || 'x2' for x2

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0 %if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
```

```
gas = 'N2';
NoLoc = 12; %number of locations
HiLoc = 5; %number of high pressure locations
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E =1; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.7; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 4000; %Maximum iterations
flips = 30;%Number of oscillations before force end
mul = 0.05;  %Convergence factor multiplier

%Convergence multipliers
con = zeros(1,NoLoc);
conf = zeros(1,NoLoc);
for i=1:NoLoc
    con(i) = 1.5;
    conf(i) = 1.5;
end

%Function switches
switch func
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
    case 0
    otherwise
        error("Incorrect input selector\n");
end

%Main Data structure. Syntax -- l(location,iteration).parameter
l = struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[]);
lfpos = zeros(1,NoLoc);
%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
%3 -- before split, after 1st exchanger (hot stream)
%4 -- before expander
%5 -- after split, before 2nd exchanger (hot stream)
```

```
%6 -- after 2nd exchanger, before valve
%7 -- after valve, before separator
%8 -- after separator (gas), before join
%9 -- after expander
%10 -- after join, before 2nd exchanger (cold)
%11 -- after 2nd exchanger(cold stream),before 1st exchanger (cold)
%12 -- after 1st exchanger (cold stream)

%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e split
f2 = f1*(1-x);
%e flow
fe = f1 * x;

for i=1:NoLoc
    l(i,1).F = fe;%initialise all as e, as they are spread
end
for i=1:3
    l(i,1).F = f1; %then correct others in groups, before split
end
for i=5:8
    l(i,1).F = f2; %after split main
end
for i=10:12
    l(i,1).F = f1; %after join
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end

%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H); %per kg
isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork

c = 2;
breaking = 0;
converged = 0;
NoFlips = 1;%Counter for number of flips [ZERO]
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
```

```
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %2nd heat exchanger  5 6 10 11

    [l(6,c+1).HP,l(11,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(10,c),l(6,c),l(11,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(11) = l(10,c).F;

    %1st heat exchanger 2 3 11 12

    [l(3,c+1).HP,l(12,c+1).HP]                                                    =
HeatExchanger11(l(2,1),l(11,c),l(3,c),l(12,c),E,NoFlips,gas);
    lfpos(3) = l(2,1).F;
    lfpos(12) = l(11,c).F;

    %Expander 4->9

    [l(9,c+1).HP,workExp] = Expander(l(4,c),l(9,c),ExpEff,gas);
    lfpos(9) = l(4,c).F;

    %Mixing streams 8&9 =10

    l(10,c+1).HP = MixedStreams3(l(8,c), l(9,c));
    lfpos(10) = l(8,c).F + l(9,c).F;

    %Split streams 3= 4&5

    l(5,c+1).HP = l(3,c).H;
    lfpos(5) = l(3,c).F * (1-x);

    l(4,c+1).HP = l(3,c).H;
    lfpos(4) = l(3,c).F * x;


    %JT expansion 6-7

    l(7,c+1).HP = l(6,c).H;
    lfpos(7) = l(6,c).F;

    %Phase Separator 7-8

    if l(7,c).Q == 1
        l(8,c+1).HP = l(7,c).H;
    else
        l(8,c+1).HP = CoolProp.PropsSI('H','P',l(8,c).P,'Q',1,gas);
    end
    lfpos(8) = l(7,c).F * l(7,c).Q;


    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
```

```
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end


    %Update Residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for each
location
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for enth
residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E==1
                NoFlips = NoFlips + 1;
            end
        end

        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for flip
            conf(i) = conf(i) + mul;
        end
    end

    %Max iterations or oscillations
    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);



            l(7,c).Q = mean([l(7,round(2*c/3):c).Q]); %After valve
            l(9,c).H = mean([l(9,round(2*c/3):c).H]); %After Expander
            l(4,c).F = mean([l(4,round(2*c/3):c).F]); %Before expander
            l(4,c).H = mean([l(4,round(2*c/3):c).H]);
            [l(9,c+1).HP,workExp] = Expander(l(4,c),l(9,c),ExpEff,gas); %before
and after expander
            breaking=1;
            break
        end
    end
    if breaking ==1
        break %also break out of while
    end
    %Covergence checks
    if c>NoLoc %After number of locations worth of iterations, start checking
for convergence
        converged = 1; %Set to yes
        maxr = 0;
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                converged = 0; %Change back if any aren't converged
            end
        end
    end
```

```
    c = c+1;
end

yield = (1-x)*(1-l(7,c).Q); %After valve

work =  ((workComp) - (workExp)) / yield;

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-[l(7,:).Q]));
    xlabel('Iteration')
    ylabel('Yield')

    %High pressure points
    figure
    plot([l(2,:).HP],'DisplayName','H2');
    hold on
    plot([l(3,:).HP],'DisplayName','H3');
    plot([l(4,:).HP],'DisplayName','H4');
    plot([l(5,:).HP],'DisplayName','H5');
    plot([l(6,:).HP],'DisplayName','H6');
    if l(2,c).P < CoolProp.Props1SI('Pcrit',gas)

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',0,gas)]);
    end
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off

    %Low pressure points
    figure
    plot([l(7,:).HP],'DisplayName','H7');
    hold on
    plot([l(8,:).HP],'DisplayName','H8');
    plot([l(9,:).HP],'DisplayName','H9');
    plot([l(10,:).HP],'DisplayName','H10');
    plot([l(11,:).HP],'DisplayName','H11');
    plot([l(12,:).HP],'DisplayName','H12');

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',0,gas)]);
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

```
function [yield,work,FOM,maxr] = SSHeylandt5(input,func)
%Heylandt cycle.
```

```
%func=...    0 for not function || 'P' for pressure || 'x' for x || 'E'
for exchanger eff ||
%   'd' for expander eff || 'c' for compressor eff || 'x2' for x2

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0%if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
gas = 'N2';
NoLoc = 12; %number of locations
HiLoc = 5; %number of high pressure locations
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E =1; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.4; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 4000; %Maximum iterations
flips = 40;%Number of oscillations before force end
mul = 0.05;  %Convergence factor multiplier

%Convergence multipliers
con = zeros(NoLoc,1);
conf = zeros(NoLoc,1);
for i=1:NoLoc
    con(i) = 1.5;
    conf(i) = 1.5;
end

%Function switches
switch func
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
    case 0
    otherwise
        error("Incorrect input selector\n");
end
```

```
%Main Data structure. Syntax -- l(location,iteration).parameter
l                                                                    =
struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[
]);
lfpos = zeros(NoLoc);
%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
%3 -- before expander
%4 -- after split, before 1st exchanger (hot stream)
%5 -- after 1st exchanger (hot), before 2nd exchanger (hot)
%6 -- after second exchanger (hot), before valve
%7 -- after valve, before separator
%8 -- after separator (gas), before 2nd exchanger (cold)
%9 -- after 2nd exchanger (cold), before split
%10 -- after expander
%11 -- after join, before 1st exchanger (cold)
%12 -- after 1st exchanger (cold)

%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e split
f2 = f1*(1-x);
%e flow
fe = f1 * x;

for i=1:NoLoc
    l(i,1).F = fe;%initialise all as e, as they are spread
end
for i=1:2
    l(i,1).F = f1; %then correct others in groups
end
for i=4:9
    l(i,1).F = f2;
end
for i=11:12
    l(i,1).F = f1;
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end
```

```
%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H);
%per kg isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork

c=2;
converged =0;
NoFlips = 1;
breaking = 0;
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %2nd heat exchanger 5 6 8 9

    [l(6,c+1).HP,l(9,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(8,c),l(6,c),l(9,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(9) = l(8,c).F;

    %1st heat exchanger 4 5 11 12

    [l(5,c+1).HP,l(12,c+1).HP]                                                   =
HeatExchanger11(l(4,1),l(11,c),l(5,c),l(12,c),E,NoFlips,gas);
    lfpos(5) = l(4,1).F;
    lfpos(12) = l(11,c).F;

    %Expander 3->10

    [l(10,c+1).HP,workExp] = Expander(l(3,c),l(10,c),ExpEff,gas);
    lfpos(10) = l(3,c).F;

    %Mixing streams 9&10=11

    l(11,c+1).HP = MixedStreams3(l(9,c), l(10,c));
    lfpos(11) = l(9,c).F + l(10,c).F;
```

```
    %Split streams

    l(4,c+1).HP = l(2,c).H;
    lfpos(4) = l(2,c).F * (1-x);

    l(3,c+1).HP = l(2,c).H;
    lfpos(3) = l(2,c).F * x;


    %JT expansion 6-7

    l(7,c+1).HP = l(6,c).H;
    lfpos(7) = l(6,c).F;

    %Seperator 7-8

    if l(7,c).Q == 1
        l(8,c+1).HP = l(7,c).H;
    else
        l(8,c+1).HP = CoolProp.PropsSI('H','P',l(8,c).P,'Q',1,gas);
    end
    lfpos(8) = l(7,c).F * l(7,c).Q;

    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end


    %enthalpy residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for
each loaction
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for
enth residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E==1
                NoFlips = NoFlips + 0.25;
            end
        end

        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for
flip
            conf(i) = conf(i) + mul;
        end
    end

    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
```

```
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);


            l(7,c).Q = mean([l(7,round(2*c/3):c).Q]); %After valve
            l(10,c).H = mean([l(10,round(2*c/3):c).H]); %After Expander
            l(3,c).F = mean([l(3,round(2*c/3):c).F]); %Before expander
            l(3,c).H = mean([l(3,round(2*c/3):c).H]);
            [l(10,c+1).HP,workExp]                                          =
Expander(l(3,c),l(10,c),ExpEff,gas); %before and after expander
            breaking=1;
            break
        end
    end

    if breaking ==1
        break %also break out of while
    end

    %Covergence checks
    if c>NoLoc %After number of locations worth of iterations, start
checking for convergence
        converged = 1; %Set to yes
        maxr = 0;
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                converged = 0; %Change back if any aren't converged
            end
        end
    end
    c = c+1;
end

yield = (1-x)*(1-l(7,c).Q);

work =  ((workComp) - (workExp)) / yield; %%

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-[l(7,:).Q]));
    xlabel('Iteration')
    ylabel('Yield')

    %High pressure points
    figure
    plot([l(2,:).HP],'DisplayName','H2');
    hold on
```

```
    plot([l(3,:).HP],'DisplayName','H3');
    plot([l(4,:).HP],'DisplayName','H4');
    plot([l(5,:).HP],'DisplayName','H5');
    plot([l(6,:).HP],'DisplayName','H6');
    if l(2,c).P < CoolProp.Props1SI('Pcrit',gas)

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',1,gas),CoolProp.Prop
sSI('H','P',l(2,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',0,gas),CoolProp.Prop
sSI('H','P',l(2,c).P,'Q',0,gas)]);
    end
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off

    %Low pressure points
    figure
    plot([l(7,:).HP],'DisplayName','H7');
    hold on
    plot([l(8,:).HP],'DisplayName','H8');
    plot([l(9,:).HP],'DisplayName','H9');
    plot([l(10,:).HP],'DisplayName','H10');
    plot([l(11,:).HP],'DisplayName','H11');
    plot([l(12,:).HP],'DisplayName','H12');

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',1,gas),CoolProp.Prop
sSI('H','P',l(1,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',0,gas),CoolProp.Prop
sSI('H','P',l(1,c).P,'Q',0,gas)]);
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

```
function [yield,work,FOM,maxr] = SSNewCycle1(input,func)
%New cycle 1.

%func=...     0 for not function || 'P' for pressure || 'x' for x || 'E' for
exchanger eff ||
%    'd' for expander eff || 'c' for compressor eff || 'y' for x2 || 'z' for
%    both

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0 %if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
gas = 'N2';
NoLoc = 15; %number of locations
HiLoc = 6; %number of high pressure locations
```

```
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E = 1 ; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.7; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency
x2 = 0.1;

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 6000; %Maximum iterations
flips = 60;%Number of oscillations before force end
mul = 0.05;  %Convergence factor multiplier

%Convergence multipliers
con = zeros(NoLoc);
conf = zeros(NoLoc);
for i=1:NoLoc
    con(i) = 1.5;
    conf(i) = 1.5;
end

%Function switches
switch func
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
    case 'y'
        x2 = input;
    case 'z'
        x = input.x1;
        x2 = input.x2;
    case 0
    otherwise
        error("Incorrect input selector\n");
end

%Main Data structure. Syntax -- l(location,iteration).parameter
l = struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[]);
lfpos = zeros(NoLoc);
%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
```

```
%3 -- before split, after 1st exchanger (hot stream)
%4 -- before expander
%5 -- after split, before 2nd exchanger (hot stream)
%6 -- after 2nd exchanger, before 3rd exchanger (hot stream)
%7 -- after 3rd exchanger (hot stream), before valve
%8 -- after valve, before separator
%9 -- after separator (gas), before join
%10 -- after split from separator (liquid), before 3rd exchanger (cold)
%11 -- after 3rd exchanger (cold), before join
%12 -- after expander, before join
%13 -- after join, before 2nd exchanger (cold)
%14 -- after 2nd exchanger (cold), before 1st exchanger (cold)
%15 -- after 1st exchanger (cold)

%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e split
f2 = f1*(1-x);
%e flow
fe = f1 * x;

for i=1:NoLoc
    l(i,1).F = fe;%initialise all as e, as they are spread
end
for i=1:3
    l(i,1).F = f1; %then correct others in groups
end
for i=5:9
    l(i,1).F = f2;
end
for i=10:11
    l(i,1).F = 0;
end
for i=13:15
    l(i,1).F = f1;
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end

%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H); %per kg
isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork

c=2;
converged =0;
NoFlips = 1;
breaking = 0;
```

```
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %3rd heat exchanger 6 7 10 11

    [l(7,c+1).HP,l(11,c+1).HP]                                                    =
HeatExchanger11(l(6,c),l(10,c),l(7,c),l(11,c),E,NoFlips,gas);
    lfpos(7) = l(6,c).F;
    lfpos(11) = l(10,c).F;

    %2nd heat exchanger 5 6 13 14

    [l(6,c+1).HP,l(14,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(13,c),l(6,c),l(14,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(14) = l(13,c).F;

    %1st heat exchanger 2 3 14 15

    [l(3,c+1).HP,l(15,c+1).HP]                                                    =
HeatExchanger11(l(2,1),l(14,c),l(3,c),l(15,c),E,NoFlips,gas);
    lfpos(3) = l(2,1).F;
    lfpos(15) = l(14,c).F;

    %Expander 4->12

    [l(12,c+1).HP,workExp] = Expander(l(4,c),l(12,c),ExpEff,gas);
    lfpos(12) = l(4,c).F;

    %Mixing streams 9&11&12=13

    l(13,c+1).HP = MixedStreams4(l(9,c), l(11,c), l(12,c));
    lfpos(13) = l(9,c).F + l(11,c).F + l(12,c).F;

    %Split streams

    l(5,c+1).HP = l(3,c).H;
    lfpos(5) = l(3,c).F * (1-x);

    l(4,c+1).HP = l(3,c).H;
    lfpos(4) = l(3,c).F * x;


    %JT expansion 7-8

    l(8,c+1).HP = l(7,c).H;
    lfpos(8) = l(7,c).F;

    %Seperator 8-9 (&10-liquid)
```

```
    if l(8,c).Q == 1
        l(9,c+1).HP = l(8,c).H;
    else
        l(9,c+1).HP = CoolProp.PropsSI('H','P',l(9,c).P,'Q',1,gas);
    end
    l(10,c+1).HP = CoolProp.PropsSI('H','P',l(10,c).P,'Q',0,gas);

    lfpos(9) = l(8,c).F * l(8,c).Q;
    lfpos(10) = (l(8,c).F * (1 - l(8,c).Q)) * x2;

    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end



    %enthalpy residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for each
location
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for enth
residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E == 1
                NoFlips = NoFlips + 0.1;
            end
        end

        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for flip
            conf(i) = conf(i) + mul;
        end
    end

    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);



            l(8,c).Q = mean([l(8,round(2*c/3):c).Q]); %After valve
            l(12,c).H = mean([l(12,round(2*c/3):c).H]); %After Expander
            l(4,c).F = mean([l(4,round(2*c/3):c).F]); %Before expander
            l(4,c).H = mean([l(4,round(2*c/3):c).H]);
            [l(12,c+1).HP,workExp]    =    Expander(l(4,c),l(12,c),ExpEff,gas);
%before and after expander
            breaking=1;
            break
```

```
        end
    end

    if breaking ==1
        break %also break out of while
    end

    %Covergence checks
    if c>NoLoc %After number of locations worth of iterations, start checking
for convergence
        converged = 1; %Set to yes
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                maxr = 1;
                converged = 0; %Change back if any aren't converged
            end
        end
    end
    c = c+1;
end

yield = (1-x)*(1-x2)*(1-l(8,c).Q);

work =  ((workComp) - (workExp)) / yield; %%

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-x2)*(1-[l(8,:).Q]));
    xlabel('Iteration')
    ylabel('Yield')



    figure
    plot([l(3,:).HP],'DisplayName','H3');
    hold on
    plot([l(4,:).HP],'DisplayName','H4');
    plot([l(5,:).HP],'DisplayName','H5');
    plot([l(6,:).HP],'DisplayName','H6');
    plot([l(7,:).HP],'DisplayName','H7');
    if l(2,c).P < CoolProp.Props1SI('Pcrit',gas)

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',0,gas)]);
    end
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off

    figure
    plot([l(8,:).HP],'DisplayName','H8');
    hold on
    plot([l(9,:).HP],'DisplayName','H9');
    plot([l(10,:).HP],'DisplayName','H10');
    plot([l(11,:).HP],'DisplayName','H11');
```

```
    plot([l(12,:).HP],'DisplayName','H12');
    plot([l(13,:).HP],'DisplayName','H13');
    plot([l(14,:).HP],'DisplayName','H14');
    plot([l(15,:).HP],'DisplayName','H15');

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',0,gas)]);
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

```
function [yield,work,FOM,maxr] = SSNewCycle2(input,func)
%New cycle 2.
%func=...     0 for not function || 'P' for pressure || 'x' for x || 'E' for
exchanger eff ||
%    'd' for expander eff || 'c' for compressor eff || 'y' for x2 || 'z'
%    for both

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0 %if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
gas = 'N2';
NoLoc = 13; %number of locations
HiLoc = 5; %number of high pressure locations
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E =1; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.7; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency
x2 = 0.1;

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 4000; %Maximum iterations
flips = 30;%Number of oscillations before force end
mul = 0.05;  %Convergence factor multiplier

%Convergence multipliers
con = zeros(NoLoc);
conf = zeros(NoLoc);
for i=1:NoLoc
    con(i) = 1.5;
    conf(i) = 1.5;
end

%Function switches
switch func
```

```
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
    case 'y'
        x2 = input;
    case 'z'
        x = input.x1;
        x2 = input.x2;
    case 0
    otherwise
        error("Incorrect input selector\n");
end

%Main Data structure. Syntax -- l(location,iteration).parameter
l = struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[]);
lfpos = zeros(NoLoc);
%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
%3 -- before split, after 1st exchanger (hot stream)
%4 -- before expander
%5 -- after split, before 2nd exchanger (hot stream)
%6 -- after 2nd exchanger (hot), before valve
%7 -- after valve, before separator
%8 -- after separator (gas), before join
%9 -- after separator (liquid) and split, before join
%10 -- after expander, before join
%11 -- after join, before 2nd exchanger (cold)
%12 -- after 2nd exchanger (cold), before 1st exchanger (cold)
%13 -- after 1st exchanger (cold)

%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e split
f2 = f1*(1-x);
%e flow
fe = f1 * x;

for i=1:NoLoc
    l(i,1).F = fe;%initialise all as e, as they are spread
end
for i=1:3
    l(i,1).F = f1; %then correct others in groups
end
for i=5:8
```

```
    l(i,1).F = f2;
end
l(9,1).F = 0;
for i=11:13
    l(i,1).F = f1;
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end

%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H); %per kg
isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork

c=2;
converged =0;
NoFlips = 1;
breaking = 0;
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %2nd heat exchanger 5 6 11 12

    [l(6,c+1).HP,l(12,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(11,c),l(6,c),l(12,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(12) = l(11,c).F;

    %1st heat exchanger 2 3 12 13

    [l(3,c+1).HP,l(13,c+1).HP]                                                    =
HeatExchanger11(l(2,1),l(12,c),l(3,c),l(13,c),E,NoFlips,gas);
    lfpos(3) = l(2,1).F;
    lfpos(13) = l(12,c).F;

    %Expander 4->10
```

```
    [l(10,c+1).HP,workExp] = Expander(l(4,c),l(10,c),ExpEff,gas);
    lfpos(10) = l(4,c).F;

    %Mixing streams 8&9&10=11

    l(11,c+1).HP = MixedStreams4(l(8,c), l(9,c), l(10,c));
    lfpos(11) = l(8,c).F + l(9,c).F + l(10,c).F;

    %Split streams

    l(5,c+1).HP = l(3,c).H;
    lfpos(5) = l(3,c).F * (1-x);

    l(4,c+1).HP = l(3,c).H;
    lfpos(4) = l(3,c).F * x;


    %JT expansion 6-7

    l(7,c+1).HP = l(6,c).H;
    lfpos(7) = l(6,c).F;

    %Seperator 7-8 (&9-liquid)

    if l(7,c).Q == 1
        l(8,c+1).HP = l(7,c).H;
    else
        l(8,c+1).HP = CoolProp.PropsSI('H','P',l(8,c).P,'Q',1,gas);
    end
    l(9,c+1).HP = CoolProp.PropsSI('H','P',l(10,c).P,'Q',0,gas);

    lfpos(8) = l(7,c).F * l(7,c).Q;
    lfpos(9) = (l(7,c).F * (1 - l(7,c).Q)) * x2;

    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end


    %enthalpy residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for each
location
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for enth
residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E == 1
                NoFlips = NoFlips + 0.5;
            end
        end

        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for flip
            conf(i) = conf(i) + mul;
```

```
        end
    end

    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);



            l(7,c).Q = mean([l(7,round(2*c/3):c).Q]); %After valve
            l(10,c).H = mean([l(10,round(2*c/3):c).H]); %After Expander
            l(4,c).F = mean([l(4,round(2*c/3):c).F]); %Before expander
            l(4,c).H = mean([l(4,round(2*c/3):c).H]);
            [l(10,c+1).HP,workExp]    =    Expander(l(4,c),l(10,c),ExpEff,gas);
%before and after expander
            breaking=1;
            break
        end
    end

    if breaking ==1
        break %also break out of while
    end

    %Convergence checks
    if c>NoLoc %After number of locations worth of iterations, start checking
for convergence
        converged = 1; %Set to yes
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                maxr = 1;
                converged = 0; %Change back if any aren't converged
            end
        end
    end
    c = c+1;
end

yield = (1-x)*(1-x2)*(1-l(7,c).Q);

work =  ((workComp) - (workExp)) / yield; %%

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-x2)*(1-[l(7,:).Q]));
    xlabel('Iteration')
    ylabel('Yield')



    figure
```

```
    plot([l(3,:).HP],'DisplayName','H3');
    hold on
    plot([l(4,:).HP],'DisplayName','H4');
    plot([l(5,:).HP],'DisplayName','H5');
    plot([l(6,:).HP],'DisplayName','H6');
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off

    figure
    plot([l(7,:).HP],'DisplayName','H7');
    hold on
    plot([l(8,:).HP],'DisplayName','H8');
    plot([l(9,:).HP],'DisplayName','H9');
    plot([l(10,:).HP],'DisplayName','H10');
    plot([l(11,:).HP],'DisplayName','H11');
    plot([l(12,:).HP],'DisplayName','H12');
    plot([l(13,:).HP],'DisplayName','H13');
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

```
%function [yield,work,FOM,maxr] = SSNewCycle3(input,func)
%New cycle 3.

%func=...     0 for not function || 'P' for pressure || 'x' for x || 'E' for
exchanger eff ||
%    'd' for expander eff || 'c' for compressor eff || 'y' for x2 || 'z' for
%    both x1 and x2

%4 changed to gradient function
%5 whole location array to tidy

if exist('func','var') == 0 || func == 0 %if not being used as a function
   func = 0;
   addpath('C:\Users\gxt357\Documents\CPMain')
   clearvars -except func %clear all variables except the func
end

%Model Parameters
gas = 'N2';
NoLoc = 18; %number of locations
HiLoc = 8; %number of high pressure locations
P1 = 101300; %Ambient Pressure [Pa]
P2 = 6000000; %Compressor outlet pressure [Pa]
T1 = 300; %Ambient Temperature [K]
T2 = 300; %Temperature at compressor outlet [K]
CompEff = 0.85; %Compressor efficiency
CompFlow = 1; %Compressor flow rate [kg/s]
E =1; %Heat exchangers effectiveness [CANNOT BE 1, USE 0.999 INSTEAD]
x = 0.65; %Fraction diverted to expander
ExpEff = 0.7; %Expander Isentropic efficiency
x2 = 0.5;

%Convergence Parameters
rcon = 1; %Convergence residual target
maxIt = 4000; %Maximum iterations
flips = 30; %Number of oscillations before force end
mul = 0.05; %Convergence factor multiplier
```

```
%Convergence multipliers
con = zeros(NoLoc,1);
conf = zeros(NoLoc,1);
for i=1:NoLoc
    con(i) = 1.4;
    conf(i) = 1.4;
end

%Function switches
switch func
    case 'P'
        P2=input;
    case 'E'
        E = input;
    case 'd'
        ExpEff = input;
    case 'c'
        CompEff = input;
    case 'x'
        x = input;
    case 'y'
        x2 = input;
    case 'z'
        x = input.x1;
        x2 = input.x2;
    case 0
    otherwise
        error("Incorrect input selector\n");
end

%Main Data structure. Syntax -- l(location,iteration).parameter
l = struct('T',[],'P',[],'H',[],'S',[],'Q',[],'F',[],'HP',[],'R',[],'RF',[]);
lfpos = zeros(NoLoc);
%Parameters:
%T -- Temperature
%P -- Pressure
%H -- Enthalpy
%S -- Entropy
%Q -- Quality
%F -- Mass flow rate
%HP-- Possible next enthalpy
%R -- Enthalpy residual
%RF-- Mass flow residual

%Locations:
%1 -- Before compressor
%2 -- After compressor
%3 -- before 1st split, after 1st exchanger (hot stream)
%4 -- before 1st expander
%5 -- after 1st split, before 2nd exchanger (hot)
%6 -- after 2nd exchanger (hot), before 2nd split
%7 -- before 2nd expander
%8 -- after 2nd split, before 3rd exchanger (hot)
%9 -- after 3rd exchanger (hot), before valve
%10 -- after valve, before separator
%11 -- after separator (gas), before 2nd join
%12 -- after 2nd expander
%13 -- after 2nd join, before 3rd exchanger (cold)
%14 -- after 3rd exchanger (cold), before 1st join
%15 -- after 1st expander
%16 -- after join, before 2nd exchanger (cold)
%17 -- after 2nd exchanger (cold), before 1st exchanger (cold)
%18 -- after 1st exchanger (cold)
```

```
%Flow Rate initialisation
%flow after compressor
f1 = CompFlow;
%flow after e1 split
f2 = f1*(1-x);
%flow after e2 split
f3 = f2*(1-x2);
%e1 flow
fe1 = f1 * x;
%e2 flow
fe2 = f2 * x2;

for i=1:NoLoc
    l(i,1).F = fe1;%initialise all as e, as they are spread
end
for i=1:3
    l(i,1).F = f1; %then correct others in groups
end
for i=5:6
    l(i,1).F = f2;
end
l(7,1).F = fe2;
for i=8:11
    l(i,1).F = f3;
end
l(12,1).F = fe2;
for i=13:14
    l(i,1).F = f2;
end
for i=16:18
    l(i,1).F = f1;
end

%All location and parameter initialising using P and T
for i=1:NoLoc
    l(i,1) = InitH2(l(i,1),P1,T1,gas);
    l(i,2) = InitH2(l(i,1),P1,T1,gas);
end

%High pressure location initialising using high P and T
for i=2:HiLoc+1 %+1 as starts on 2
    l(i,1) = InitH2(l(i,1),P2,T2,gas);
    l(i,2) = InitH2(l(i,1),P2,T2,gas);
end

%Compressor calculation
isothermwork= (l(1,1).T * (l(1,1).S - l(2,1).S)) - (l(1,1).H-l(2,1).H); %per kg
isothermal work

H2isen = CoolProp.PropsSI('H','S',l(1,1).S,'P',l(2,1).P,gas);
isentropwork= abs(H2isen - l(1,1).H);

workComp  = isentropwork/CompEff;%isentropwork

c=2;
converged =0;
NoFlips = 1;
breaking = 0;
%try
while (converged==0)

    if func == 0
        fprintf('%d\n',c);
```

```
    end
    %Compressor and ambient
    l(1,c+1).HP = l(1,c).H;
    lfpos(1) = l(1,c).F;

    l(2,c+1).HP = l(2,c).H;
    lfpos(2) = l(2,c).F;

    %3rd heat exchanger 8 9 13 14

    [l(9,c+1).HP,l(14,c+1).HP]                                                    =
HeatExchanger11(l(8,c),l(13,c),l(9,c),l(14,c),E,NoFlips,gas);
    lfpos(9) = l(8,c).F;
    lfpos(14) = l(13,c).F;

    %2nd heat exchanger 5 6 16 17

    [l(6,c+1).HP,l(17,c+1).HP]                                                    =
HeatExchanger11(l(5,c),l(16,c),l(6,c),l(17,c),E,NoFlips,gas);
    lfpos(6) = l(5,c).F;
    lfpos(17) = l(16,c).F;

    %1st heat exchanger 2 3 17 18

    [l(3,c+1).HP,l(18,c+1).HP]                                                    =
HeatExchanger11(l(2,1),l(17,c),l(3,c),l(18,c),E,NoFlips,gas);
    lfpos(3) = l(2,1).F;
    lfpos(18) = l(17,c).F;

    %1st Expander 4->15

    [l(15,c+1).HP,workExp1] = Expander(l(4,c),l(15,c),ExpEff,gas);
    lfpos(15) = l(4,c).F;

    %1st Mixing streams 14&15=16

    l(16,c+1).HP = MixedStreams3(l(14,c), l(15,c));
    lfpos(16) = l(14,c).F + l(15,c).F;

    %1st Split streams 3 4 5

    l(5,c+1).HP = l(3,c).H;
    lfpos(5) = l(3,c).F * (1-x);

    l(4,c+1).HP = l(3,c).H;
    lfpos(4) = l(3,c).F * x;

    %2nd Expander 7-12
    [l(12,c+1).HP,workExp2] = Expander(l(7,c),l(12,c),ExpEff,gas);
    lfpos(12) = l(7,c).F;

    %2nd Mixing streams 11&12=13

    l(13,c+1).HP = MixedStreams3(l(11,c), l(12,c));
    lfpos(13) = l(11,c).F + l(12,c).F;

    %2nd Split streams 6 7 8

    l(8,c+1).HP = l(6,c).H;
```

```
    lfpos(8) = l(6,c).F * (1-x2);

    l(7,c+1).HP = l(6,c).H;
    lfpos(7) = l(6,c).F * x2;


    %JT expansion 9-10

    l(10,c+1).HP = l(9,c).H;
    lfpos(10) = l(9,c).F;

    %Seperator 8-9 10-11

    if l(10,c).Q == 1
        l(11,c+1).HP = l(10,c).H;
    else
        l(11,c+1).HP = CoolProp.PropsSI('H','P',l(11,c).P,'Q',1,gas);
    end
    lfpos(11) = l(10,c).F * l(10,c).Q;

    %Modifying/updating for convergence
    for i=1:NoLoc
        Hpos = UpdateGrad(l(i,c-1),l(i,c),l(i,c+1));
        l(i,c+1).H = l(i,c).H - ((l(i,c).H- Hpos)/con(i));
        l(i,c+1).P = l(i,c).P;
        l(i,c+1) = HPUpdate2(l(i,c+1),gas);
        l(i,c+1).F = l(i,c).F - ((l(i,c).F- lfpos(i))/conf(i));
    end


    %enthalpy residuals

    for i=1:NoLoc
        l(i,c).R = l(i,c).H - l(i,c).HP; %update enthalpy residual for each
location
        if (l(i,c+1).H-l(i,c).H) * (l(i,c).H-l(i,c-1).H) < 0 %check for enth
residual flip?
            con(i) = con(i) + mul; %increase convergence multiplier
            if E == 1
                NoFlips = NoFlips + 0.1;
            end
        end

        l(i,c).RF = l(i,c).F - lfpos(i); %update flow rate residual
        if (l(i,c+1).F-l(i,c).F) * (l(i,c).F-l(i,c-1).F) < 0 %check for flip
            conf(i) = conf(i) + mul;
        end
    end

    for i=3:NoLoc

        if con(i) >= flips || c > maxIt
            if c > maxIt
                fprintf('Max iterations reached\n');
            else
                fprintf('Oscillating\n');
            end

            maxr = round(max(abs([l(:,c).R])),0);
            fprintf('(Max residual = %.0f)\n',maxr);
```

```
            l(10,c).Q = mean([l(10,round(2*c/3):c).Q]); %After valve
            l(15,c).H = mean([l(15,round(2*c/3):c).H]); %After Expander 1
            l(4,c).F = mean([l(4,round(2*c/3):c).F]); %Before expander 1
            l(4,c).H = mean([l(4,round(2*c/3):c).H]);
            [l(15,c+1).HP,workExp1]    =    Expander(l(4,c),l(15,c),ExpEff,gas);
%before and after expander

            l(12,c).H = mean([l(12,round(2*c/3):c).H]); %After Expander 2
            l(7,c).F = mean([l(7,round(2*c/3):c).F]); %Before expander 2
            l(7,c).H = mean([l(7,round(2*c/3):c).H]);
            [l(12,c+1).HP,workExp2]    =    Expander(l(7,c),l(12,c),ExpEff,gas);
%before and after expander
            breaking=1;
            break
        end
    end

    if breaking ==1
        break %also break out of while
    end

    %Convergence checks
    if c>NoLoc %After number of locations worth of iterations, start checking
for convergence
        converged = 1; %Set to yes
        for i=1:NoLoc
            if abs(l(i,c).R) > rcon
                maxr = 1;
                converged = 0; %Change back if any aren't converged
            end
        end
    end
    c = c+1;
end

yield = (1-x)*(1-x2)*(1-l(10,c).Q);

work =  ((workComp) - (workExp1 + workExp2)) / yield; %%

workIdeal =IdealLiqueWork2(P1,T1,gas);
FOM = workIdeal/work;

if func ==0
    fprintf('Yield = %f\nFOM = %.2f%%\n',yield,FOM*100);
    figure
    plot((1-x)*(1-x2)*(1-[l(10,:).Q]));
    xlabel('Iteration')
    ylabel('Yield')



    figure
    plot([l(3,:).HP],'DisplayName','H3');
    hold on
    plot([l(4,:).HP],'DisplayName','H4');
    plot([l(5,:).HP],'DisplayName','H5');
    plot([l(6,:).HP],'DisplayName','H6');
    plot([l(7,:).HP],'DisplayName','H7');
    plot([l(8,:).HP],'DisplayName','H8');
    plot([l(9,:).HP],'DisplayName','H9');
    if l(2,c).P < CoolProp.Props1SI('Pcrit',gas)
```

```
line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(2,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(2,c).P,'Q',0,gas)]);
    end
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off

    figure
    plot([l(10,:).HP],'DisplayName','H10');
    hold on
    plot([l(11,:).HP],'DisplayName','H11');
    plot([l(12,:).HP],'DisplayName','H12');
    plot([l(13,:).HP],'DisplayName','H13');
    plot([l(14,:).HP],'DisplayName','H14');
    plot([l(15,:).HP],'DisplayName','H15');
    plot([l(16,:).HP],'DisplayName','H16');
    plot([l(17,:).HP],'DisplayName','H17');
    plot([l(18,:).HP],'DisplayName','H18');

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',1,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',1,gas)]);

line([0,c],[CoolProp.PropsSI('H','P',l(1,c).P,'Q',0,gas),CoolProp.PropsSI('H',
'P',l(1,c).P,'Q',0,gas)]);
    xlabel('Iteration')
    ylabel('H [J]')
    legend('show')
    hold off
end
```

Appendix B: User Defined Functions - Standard Wankel Expander

```
#include "udf.h"
#include <math.h>
#define E 0.004125 /*Rotor eccentricity [mm]*/
#define R 0.030 /*Rotor radius [mm]*/
#define RPM 4800 /*Crankshaft speed [rpm]*/
#define C 0.00005 /*Rotor/Housing clearance [mm]*/


/*START OF HOUSING UDF*/

/*find positive arc tangent*/
static real my_atan(real y, real x)
{
	real result;
	result = atan2(y, x);
	if (atan2(y, x)<0)
		result = atan2(y, x) + 2 * M_PI;
	return result;
}

/*This function takes a current node's (x,y) coordinate and the time-step size as
inputs and updates the new node (x,y) coordinates*/
static void findnew(real* pp1, real* pp2, real p1, real p2, real dtime)
{
	real cosa; /*cos(alpha) value*/
	real alpha; /*alpha angle*/
	real beta; /*beata angle*/
	real theta; /*theta angle*/

	cosa = ((E * E) + ((R+C) * (R+C)) - (p1 * p1) - (p2 * p2)) / (2 * E*(R+C));
/*equation of triangle rearranged for cos(a)*/

	/*prevents erroneous cosa readings*/
	if (cosa > 1)
	{
		beta = 0;
	}
	else if (cosa < -1)
	{
		beta = M_PI;
	}
	else
	{
		beta = acos(cosa);
	}

	/* find alpha, which is angle of node coord from +x plane*/
	alpha = my_atan(p2, p1);
```

```
	/*find which quadrant the node is in and then find angle theta from angle
beta (the angle between E and R)*/
	if (alpha > 0 && alpha < M_PI / 2)
	{
		theta = (M_PI - beta) / 2;
	}
	else if (alpha > M_PI && alpha < 3 * M_PI / 2)
	{
		theta = (3 * M_PI - beta) / 2;
	}
	else if (alpha > M_PI / 2 && alpha < M_PI)
	{
		theta = (M_PI + beta) / 2;
	}
	else if (alpha > 3 * M_PI / 2 && alpha < 2 * M_PI)
	{
		theta = (3 * M_PI + beta) / 2;
	}
	else if (alpha == 0 || alpha == M_PI || alpha == M_PI/2 || alpha == 3*M_PI/2)
	{
		theta = alpha;
	}
	else
	{
		Message("\nNOTE- Cannot determine quadrant ( alpha = %f ) Asumming
theta = 0\n", alpha);
		theta = 0;
	}

	theta = theta + (dtime*RPM*M_PI / 90);/*adds the amount of radians rotated
by the rotor in one timestep to the current theta (so rotor apex motion matches
housing mesh motion)*/

	*pp1 = E*cos(theta * 3) + (R+C)*cos(theta); /*calculate nodes new coordinates
from new theta and update*/
	*pp2 = E*sin(theta * 3) + (R+C)*sin(theta);
}

/*This is the UDF macro that cycles through all nodes in the housing mesh domain*/
DEFINE_GRID_MOTION(housing, domain, dt, time, dtime)
{
	real p1, p2, p3; /*current position*/

	Thread *tf = DT_THREAD(dt);
	face_t f;
	int n;
	Node *v;
	SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

	begin_f_loop(f, tf) /*loop through faces in the mesh domain*/
	{
		f_node_loop(f, tf, n) /*loop through nodes in the face*/
		{
			v = F_NODE(f, tf, n); /*Assign node identity to v*/

			if (NODE_POS_NEED_UPDATE(v)) /*if the node hasn't been updated
already*/
			{
				NODE_POS_UPDATED(v); /*set node to flag as updated*/
				/*find current node coordinates*/
```

```
                        p1 = NODE_X(v);
                        p2 = NODE_Y(v);
                        p3 = NODE_Z(v);

                        findnew(&p1, &p2, p1, p2, dtime); /*Update coordinates
function*/

                        NODE_X(v) = p1; /*assign new node coordinates*/
                        NODE_Y(v) = p2;

                }
            }
        }
        end_f_loop(f, tf);
}


/* START OF Rotor UDF*/

/*find positive arc tangent*/
static real my_atan2(real x, real y)
{
        real result;

        result = atan2(y, x);
        if (atan2(y, x)<0)
                result = atan2(y, x) + 2 * M_PI;

        return result;
}

/*This function rotates the nodes around the rotor centre axis*/
static void Rotate(real* ppx, real* ppy, real px, real py, real drangle, real CGx,
real CGy)
{
        real theta, rad;

        theta = my_atan2(px - CGx, py - CGy); /*current angle of node from around
rotor cnetre axis*/
        rad = sqrt(SQR(px - CGx) + SQR(py - CGy)); /*distance of node from rotor
centre axis*/
        theta = theta + drangle; /*time-step updated theta angle*/

        /*Update node coordinates*/
        *ppx = CGx + (rad*cos(theta));
        *ppy = CGy + (rad*sin(theta));
}


/*This is the UDF macro that cycles through all nodes in the rotor mesh domain*/
DEFINE_GRID_MOTION(Rotor, domain, dt, time, dtime)
{
        real px, py, pz, rangle, eangle, drangle, deangle;/*px,py,pz=current node
coordinates; eangle= eccentric angle; rangle= rotor angle; deangle and drangle are
the angle steps*/
        real dx, dy, CGx, CGy, CGxnew, CGynew; /*dx,dy=change in x and y coordinates;
CGx,CGY=rotor current centre axis coords; CGxnew,CGynew=updated centre axis
coords*/

        eangle = (RPM * M_PI*time / 30) - (M_PI / 2); /*eccentric angle*/
```

```
        deangle = RPM * M_PI*dtime / 30; /*eccentric angle step*/
        drangle = RPM * M_PI*dtime / 90; /*rotor angle step*/

        CGx = E * cos(eangle - deangle); /*find previous time-step CG coordinates*/
        CGy = E * sin(eangle - deangle);

        CGxnew = E * cos(eangle); /*find this time-step CG coords*/
        CGynew = E * sin(eangle);

        dx = CGxnew - CGx; /*find change in CG coords*/
        dy = CGynew - CGy;

        Thread *tf = DT_THREAD(dt);
        face_t f;
        int n;
        Node *v;
        SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

        begin_f_loop(f, tf) /*loop faces in mesh domain*/
        {
               f_node_loop(f, tf, n) /*loop nodes in face*/
               {
                      v = F_NODE(f, tf, n);

                      if (NODE_POS_NEED_UPDATE(v)) /*if node not updated*/
                      {
                             NODE_POS_UPDATED(v); /*set node to updated*/
                             /*find current node coordinates*/
                             px = NODE_X(v);
                             py = NODE_Y(v);
                             pz = NODE_Z(v);

                             /*Translate rotor centre axis (CG)*/
                             px = px + dx;
                             py = py + dy;

                             /*Rotate about CG*/
                             Rotate(&px, &py, px, py, drangle, CGxnew, CGynew);

                             NODE_X(v) = px; /*assign new node coordinates*/
                             NODE_Y(v) = py;
                      }
               }
        }
        end_f_loop(f, tf);
}
```

Appendix C: User Defined Functions - Static Shaft Wankel Expander

```
#include "udf.h"
#include <math.h>
#define E 0.004125 /*Rotor eccentricity [mm]*/
#define R 0.030 /*Rotor Radius [mm]*/
#define RPM 4800 /*Crankshaft speed [rpm]*/
#define C 0.0001 /*Clearance gap [mm]*/

static void Rotateback(real* ppx, real* ppy, real px, real py, real time)
{
	real theta, rad,dangle;

	dangle = ((2 * M_PI * RPM) / 60)*time; /*rotation from 0 time in s*/

	theta = my_atan2(px, py); /*angle from origin*/
	rad = sqrt(SQR(px) + SQR(py)); /*distance from origin*/
	theta = theta - dangle;

	/*Update coordinates */
	*ppx = rad*cos(theta);
	*ppy = rad*sin(theta);
}
static void Rotateforward(real* ppx, real* ppy, real px, real py, real time)
{
	real theta, rad, dangle;

	dangle = ((2 * M_PI * RPM) / 60)*time; /*angle of rotation from 0 time in
s*/

	theta = my_atan2(px, py); /*angle from origin*/
	rad = sqrt(SQR(px) + SQR(py)); /*distance from origin*/
	theta = theta + dangle; /*updated theta*/

	/*Update coordinates*/
	*ppx = rad*cos(theta);
	*ppy = rad*sin(theta);
}

/*START OF HOUSING UDF*/
/*finds positive arc tangent*/
static real my_atan(real y, real x)
{
	real result;
	result = atan2(y, x);
	if (atan2(y, x)<0)
		result = atan2(y, x) + 2 * M_PI;
	return result;
}
```

```
static void findnew(real* pp1, real* pp2, real p1, real p2, real dtime)
{
	real cosa;
	real alpha;
	real beta;
	real theta;

	cosa = ((E * E) + ((R+C) * (R+C)) - (p1 * p1) - (p2 * p2)) / (2 * E*(R+C));
/*equation of triangle rearranged for cos(a)*/

	/*prevents erroneous cosa readings*/
	if (cosa > 1)
	{
		beta = 0;
	}
	else if (cosa < -1)
	{
		beta = M_PI;
	}
	else
	{
		beta = acos(cosa);
	}

	/* find alpha, which is angle of node coord from +x plane*/
	alpha = my_atan(p2, p1);

	/*find which quadrant the node is in and find theta from angle beta (angle
between E and R)*/
	if (alpha > 0 && alpha < M_PI / 2)
	{
		theta = (M_PI - beta) / 2;
	}
	else if (alpha > M_PI && alpha < 3 * M_PI / 2)
	{
		theta = (3 * M_PI - beta) / 2;
	}
	else if (alpha > M_PI / 2 && alpha < M_PI)
	{
		theta = (M_PI + beta) / 2;
	}
	else if (alpha > 3 * M_PI / 2 && alpha < 2 * M_PI)
	{
		theta = (3 * M_PI + beta) / 2;
	}
	else if (alpha == 0 || alpha == M_PI || alpha == M_PI/2 || alpha == 3*M_PI/2)
	{
		theta = alpha;
	}
	else
	{
		Message("\nNOTE- Cannot determine quadrant ( alpha = %f ) Assuming
theta = 0\n", alpha);
		theta = 0;
	}

	theta = theta - (dtime*RPM*M_PI / 90);/*adds the amount of radians rotated
by the rotor in one time-step to the current theta (so apex motion match housing
mesh motion)*/
```

```
	*pp1 = E*cos(theta * 3) + (R+C)*cos(theta); /*calculate nodes new coordinates
from new theta*/
	*pp2 = E*sin(theta * 3) + (R+C)*sin(theta);
}

DEFINE_GRID_MOTION(housing, domain, dt, time, dtime)/*dtime= time-step size*/
{
	real p1, p2, p3; /*current position*/

	Thread *tf = DT_THREAD(dt);
	face_t f;
	int n;
	Node *v;
	SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

	begin_f_loop(f, tf)/*loop through faces in mesh domain*/
	{

		f_node_loop(f, tf, n) /*loop through nodes in face*/
		{

			v = F_NODE(f, tf, n);

			if (NODE_POS_NEED_UPDATE(v))
			{
				NODE_POS_UPDATED(v);
				/*find current node coordinates*/
				p1 = NODE_X(v);
				p2 = NODE_Y(v);
				p3 = NODE_Z(v);

				Rotateback(&p1, &p2, p1, p2, time-dtime);
/*rotate node back to starting coordinates(@ time=0) from previous position*/

				findnew(&p1, &p2, p1, p2, dtime);
/*perfrom housing node movement*/

				Rotateforward(&p1, &p2, p1, p2, time);
/*rotate node forward to new current time position*/

				NODE_X(v) = p1; /*assign new node coordinates*/
				NODE_Y(v) = p2;

			}
		}
	}
	end_f_loop(f, tf);
}

/* START OF Rotor UDF*/

static void Rotate(real* ppx, real* ppy, real px, real py, real drangle, real CGx,
real CGy)
{
	real theta, rad;

	theta = my_atan2(px - CGx, py - CGy); /*angle of node from rotor centre
axis*/
	rad = sqrt(SQR(px - CGx) + SQR(py - CGy)); /*distance of node from rotor
centre axis*/
```

```
        theta = theta + drangle; /*updated theta angle*/

        /*Update coordinates*/
        *ppx = CGx + (rad*cos(theta));
        *ppy = CGy + (rad*sin(theta));
}


DEFINE_GRID_MOTION(eccentricnew, domain, dt, time, dtime)
{
        real px, py, pz, drangle;/*px,py,pz=current position; drangle= rotor angle
step (in 1 time-step)*/

        drangle = RPM * M_PI*dtime / 45; /*angle step*/

        Thread *tf = DT_THREAD(dt);
        face_t f;
        int n;
        Node *v;
        SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

        begin_f_loop(f, tf) /*loop through faces in mesh domain*/
        {
                f_node_loop(f, tf, n) /*loop through nodes of face*/
                {
                        v = F_NODE(f, tf, n); /*assign node identity to v*/
                        if (NODE_POS_NEED_UPDATE(v))
                        {
                                NODE_POS_UPDATED(v);
                                /*find current node coordinates*/
                                px = NODE_X(v);
                                py = NODE_Y(v);
                                pz = NODE_Z(v);

                                /*Rotate geometry about rotor centre axis*/
                                Rotate(&px, &py, px, py, drangle, 0, -1 * E);

                                NODE_X(v) = px; /*assign new node coordinates*/
                                NODE_Y(v) = py;
                        }
                }
        }
        end_f_loop(f, tf);
}

/*START of outlet port UDF*/
DEFINE_GRID_MOTION(exhaustRotate, domain, dt, time, dtime)
{
        real px, py, pz, drangle;/*px,py,pz=current node coordinates. drangle= rotor
angle step (in 1 time-step)*/

        drangle = RPM * M_PI*dtime / 30; /*calculate angle step*/

        Thread *tf = DT_THREAD(dt);
        face_t f;
        int n;
        Node *v;
        SET_DEFORMING_THREAD_FLAG(THREAD_T0(tf));

        begin_f_loop(f, tf) /*loop through faces of mesh domain*/
```

```
        {
                f_node_loop(f, tf, n) /*loop through nodes of face*/
                {
                        v = F_NODE(f, tf, n); /*asigned node identity to v*/
                        if (NODE_POS_NEED_UPDATE(v))
                        {
                                NODE_POS_UPDATED(v);
                                /*find current node coordinates*/
                                px = NODE_X(v);
                                py = NODE_Y(v);
                                pz = NODE_Z(v);

                                /*Rotate geometry about origin*/
                                Rotate(&px, &py, px, py, drangle, 0, 0);

                                NODE_X(v) = px; /*assign new node coordinates*/
                                NODE_Y(v) = py;
                        }
                }
        }
        end_f_loop(f, tf);
}
```

Appendix D: MATLAB Code - CFD Post-Processing

```
%Read and process the data output files from fluent, calculating the
power,
%flow rate and efficiency of the case.
clearvars;

RPM = 4800;
rps = RPM/60;
CaseSourceFolder                                                    =
'C:\Users\gxt357\Documents\WankelCFD\StaticShaft\+10Deg\8400\4bar\0.05
2d';
FileNameFlow = '\inletmf.out';
FileNamePressure = '\pressure.out';
FileNameVolume = '\volume.out';
LocFlow = strcat(CaseSourceFolder,FileNameFlow); %location the flow file
LocPressure = strcat(CaseSourceFolder,FileNamePressure);
LocVolume = strcat(CaseSourceFolder,FileNameVolume);

%Flow rate calculation
%open flow rate output file and copy to memory
fid = fopen(LocFlow);
f                           =                           textscan(fid,'%f
%f','HeaderLines',3,'Delimiter',',','CollectOutput',1);
f = f{1};
q                           =                           textscan(fid,'%f
%f','HeaderLines',3,'Delimiter',',','CollectOutput',1);
q = q{1};
fclose(fid);

FlowTot = 0;
n=0;
for  i=1:length(f)  %sum  all  flowrate  values  for  the  last  third  of
simulation time
    if f(i,1) >= 400
    FlowTot = FlowTot + f(i,2);
    n=n+1;
    end
end
FlowAve = FlowTot / n; %calculate average flow rate

%Presure-Volume calculations
%open pressure output file and copy to memory
fid = fopen(LocPressure);
p1            =            textscan(fid,'%f            %f            %f
%f','HeaderLines',3,'Delimiter',',','CollectOutput',1);
p1 = p1{1};
fclose(fid);

%open volume output file and copy to memory
fid = fopen(LocVolume);
```

```
v1            =            textscan(fid,'%f            %f            %f
%f','HeaderLines',3,'Delimiter',',','CollectOutput',1);
v1 = v1{1};
fclose(fid);

p=zeros(600,1);
v=zeros(600,1);

%Collect pressure values from last third of each chamber's cycle
i=1;
for j=1:length(p1)
        if p1(j,1) > 400
            p(i) = p1(j,2);
            i=i+1;
        end
end
for j=1:length(p1)
        if p1(j,1) > 400
            p(i) = p1(j,3);
            i=i+1;
        end
end
for j=1:length(p1)
        if p1(j,1) > 400
            p(i) = p1(j,4);
            i=i+1;
        end
end

%Collect volume values from last third of each chamber's cycle
i=1;
for j=1:length(v1)
        if v1(j,1) > 400
            v(i) = v1(j,2);
            i=i+1;
        end
end
for j=1:length(v1)
        if v1(j,1) > 400
            v(i) = v1(j,3);
            i=i+1;
        end
end
for j=1:length(v1)
        if v1(j,1) > 400
            v(i) = v1(j,4);
            i=i+1;
        end
end

work = trapz(v,p); %Calculate area inside pressure-volume diagram
power = work * rps * 2; %calculate power output
fprintf('FlowAve = %.6e\n',FlowAve);
fprintf('Power = %.6e\n',power);
```

Appendix E: MATLAB Code -

Experimental Results Post-Processing

```
%Takes data from multiple files, matches times(1 pico file), averages
them and places them in a single
%file
clearvars;

CaseSourceFolder  =  'M:\Experimental';  %experimental  results  parent
folder
expDate = '200320'; %experiment date (DDMMYY)
expSeries = ' 4.'; %experiment series of that day

n = 8; %number of experiments in series

fileType = '.csv';
locLC1 = '\LoadCellResults\'; %Arduino results folder
locPico1 = '\PicoResults\'; %pico results folder

lcTorq = zeros(n,1);
lcFreq = zeros(n,1);
lcRPM = zeros(n,1);
picoTa = zeros(n,1);
picoTin = zeros(n,1);
picoTout = zeros(n,1);
picoPin = zeros(n,1);
picoPout = zeros(n,1);
picoDp = zeros(n,1);

for j = 1:n %loop through all results in the given series

    expNumber = num2str(j);

    %---Arduino Results---%
    locLC2                                                              =
strcat(CaseSourceFolder,locLC1,expDate,expSeries,expNumber,fileType);
%Load cell result specific file location

    %open file and copy results
    fid = fopen(locLC2);
    lc      =      textscan(fid,'%f      %f      %f      %f      %f      %f
%f','HeaderLines',0,'Delimiter',',','CollectOutput',1);
    lc = lc{1,1};
    fclose(fid);

    %average all results from file over time
    lcMean = mean(lc);
```

```
    lcTime = datetime(lc(1,1),'ConvertFrom','datenum'); %record time of
reading
    lcTorq(j,1) = lcMean(5); %record torque reading to results
    lcFreq(j,1) = lcMean(6); %record flow meter frequency reading to
results
    lcRPM(j,1) = lcMean(7); %record speed reading to results

    %----Pico results----%

    locPico2                                                          =
strcat(CaseSourceFolder,locPico1,expDate,'Pico',expSeries,'0',fileType
); %Pico result specific file location
    %open pico results file
    fid = fopen(locPico2);
    pico      =      textscan(fid,'%q      %q      %q      %q      %q
%q','HeaderLines',1,'Delimiter',',','CollectOutput',1);  %copy  results
to memory
    pico = pico{1,1}; %changes to regular cell array
    pico = str2double(pico); %converts to double array
    pico = pico(isfinite(pico(:, 1)), :); %removes blank rows
    pico = pico(isfinite(pico(:, 2)), :); %removes blank rows
    pico = pico(isfinite(pico(:, 3)), :); %removes blank rows
    pico = pico(isfinite(pico(:, 4)), :); %removes blank rows
    pico = pico(isfinite(pico(:, 5)), :); %removes blank rows
    pico = pico(isfinite(pico(:, 6)), :); %removes blank rows
    fclose(fid);

    picoTime = datetime(pico(:,1),'ConvertFrom','posixtime'); %Converts
to datetime format from pico's epoch

    picoExpStart = find(picoTime > lcTime); %find times after arduino
start to match results
    if isempty(picoExpStart) %check if empty (if arduino started before
first pico recording)
        picoExpStart = 1; %if so, start at 1st data value
    else
        picoExpStart = picoExpStart(1); %else select first time
    end

    picoExpEnd = find(picoTime > (lcTime + seconds(20))); %find times
20s after arduino start
    if isempty(picoExpEnd) %check if empty (lc finished after last pico
recording)
        picoExpEnd = length(pico); %if so use last pico values
    else
        picoExpEnd = picoExpEnd(1); %select end time
    end

    picoMeanExp  =  mean(pico(picoExpStart:picoExpEnd,:));  %finds  mean
values for the time of current experiment

    %record mean values to results output
    picoTa(j,1) = picoMeanExp(2);
    picoTin(j,1) = picoMeanExp(3);
    picoTout(j,1) = picoMeanExp(4);
    picoPin(j,1) = picoMeanExp(5);
    picoPout(j,1) = picoMeanExp(6);
```

```
end
%output all values to correctly formatted matrix, to be put into excel
file
dataOut(:,1) = lcTorq;
dataOut(:,2) = lcRPM;
dataOut(:,3) = picoTa;
dataOut(:,4) = picoTin;
dataOut(:,5) = picoTout;
dataOut(:,6) = picoPin;
dataOut(:,7) = picoPout;
dataOut(:,8) = lcFreq;
```

Appendix F: Arduino Code - Load Cell, Speed and Flow Meter Output Logging

```
#include "HX711.h"
#define samples 500
//tacho pin
//int tachoPin = 2;

// HX711 circuit wiring
const int LOADCELL_DOUT_PIN = 4;
const int LOADCELL_SCK_PIN = 5;
int rpm;
int rpmAve[10];

HX711 scale;

unsigned long findFreq() {

  int potValue;
  int i;
  int arr[samples];

  unsigned long t[samples];
  unsigned long count;
  unsigned long period;
  unsigned long aveT;
  unsigned long tPrev;
  unsigned long t0;
  unsigned long t1;

  for (i = 0; i < samples; i++) {
    arr[i] = analogRead(A15);
    t[i] = micros();
  }

  count = 0.0;
  aveT = 0.0;

  for (i = 1; i < samples; i++) {
    if (arr[i] > 0 && arr[i - 1] == 0) { //loop through array and
select each location where voltage starts increasing from zero
      count = count + 1.0; // add to count
      if (count > 1) {
        aveT = aveT + (t[i] - tPrev); //if not first count, calculate
the period between this and previous time
      }

      if (count == 1) {
        t0 = t[i]; //record first count timestamp
      }

      t1 = t[i]; //record last count timestamp
```

```
      tPrev = t[i]; // record time stamp as previous for next
iteration
    }
  }

  if (count == 0) {
    return 0.00; //zero flow case
  }

  aveT = aveT / (count - 1.0); //average totalled periods
  period = (t1 - t0) / (count - 1.0); // time from first count to last
divided by number of counts

  return 1000000.00 / period; //frequency 2
}

void setup() {
  Serial.begin(9600); delay(10);

  scale.begin(LOADCELL_DOUT_PIN, LOADCELL_SCK_PIN);

  scale.set_scale(2280.f);                      // this value is
obtained by calibrating the scale with known weights; see the README
for details

  Serial.print("Time[ms]");
  Serial.print(",");
  Serial.print("LoadCell");
  Serial.print(",");
  Serial.print("Force[N]");
  Serial.print(",");
  Serial.print("Torque[Nmm]");
  Serial.print(",");
  Serial.println("Frequency[Hz]");

}

void loop() {
  int rpmTot = 0;

  float  i = scale.get_units(); //raw load cell data - averaged as
number of readings
  float N = (0.059 * i) - 0.4703; //<--5kg || (0.2013*i) + 1.6037;
//<--20kg //calibrated to newtons 1.1037
  float T = N * 0.047 * 1000; //torque conversion was 45

  unsigned long freq = findFreq(); //Find frequency of flow meter

  rpm = analogRead(A0); //Read RPM voltage in
  //for(int j=1;j<10;j++){
  //  rpmAve[j-1]=rpmAve[j];
  //}
  //rpmAve[9] = rpm;
  //for(int j=0;j<10;j++){
  //   rpmTot = rpmTot+rpmAve[j];
  //}
  if (rpm != 0) {
    rpm = 29.94 + 3.8351 * rpm;
```

```
  }

  Serial.print(millis());
  Serial.print(",");
  Serial.print(i);
  Serial.print(",");
  Serial.print(N);
  Serial.print(",");
  Serial.print(T);
  Serial.print(",");
  Serial.print(freq);
  Serial.print(",");
  Serial.println(rpm);
}
```

Appendix G: Arduino Code - Brushless DC Motor Control

```
/*
        Arduino Brushless Motor Control
     by Dejan, https://howtomechatronics.com
*/
#include <Servo.h>
Servo ESC;     // create servo object to control the ESC
int potValue;  // value from the analogue pin
void setup() {
  // Attach the ESC on pin 9
  ESC.attach(9, 1000, 2000); // (pin, min pulse width, max pulse width
in milliseconds)
}
void loop() {
  potValue = analogRead(A0);   // reads the value of the potentiometer
(value between 0 and 1023)
  potValue = map(potValue, 0, 1023, 0, 180);   // scale it to use it
with the servo library (value between 0 and 180)
  ESC.write(potValue);    // Send the signal to the ESC
}
```

Appendix H: Flow Meter Calibration Certificates

Calibration Report
Omega Engineering
One Omega Drive
Stamford, CT 06907-0047
USA

MEAN TOTAL PULSES (PER WEIGHT USED): 3178.00
SENSING ELEMENT CONSTANT (MEAN "K"): 13926.709

Remarks:
FINAL CAL

Conversion Details: ACF (Volumetric Air Equiv)
Air

Range Bars: +/-1.00%

CALIBRATED BY:
CERTIFIED BY:
DATE: 09/24/14 Form Rev: 1.5.1

Certificate

We certificate that

Flow Meter

With the following specifications

Model	BGFT-25-NC-025-S-S-S-N-D16-N		
Serial No.	G25KMDZ009		
Nominal Range	3-30m³/h	Connection	Flange
Nominal Size	DN25	Max. Pressure	1.6MPa
Accuracy Class	1.5%	Max. Fluid Temp.	120°C

Has been tested and verified on this date: 2020-02-21

Reported Values

K-Factor:

Calibration Point	Actual Flow (m³/h)	Flow Meter Value (m³/h)	Error
1	2.93	2.89	-1.36%
2	14.69	14.91	1.48%
3	30.34	30.47	-1.11%
Conclusion:			
Actual Accuracy	1.5	Repeatability	0.13

Operating Conditions at the Flow Laboratory

Medium	Gas	Medium Temp.	5°C
Operating Pressure	101.2KPa	Ambient Temp.	10°C

Bell Flow Systems

Company:

Figure H-1 - Calibration certificates for Omega FTB-933 (left) and BGFT-25 (Right) flow meters

Appendix I: Static Shaft Wankel Expander Prototype CAD Drawings

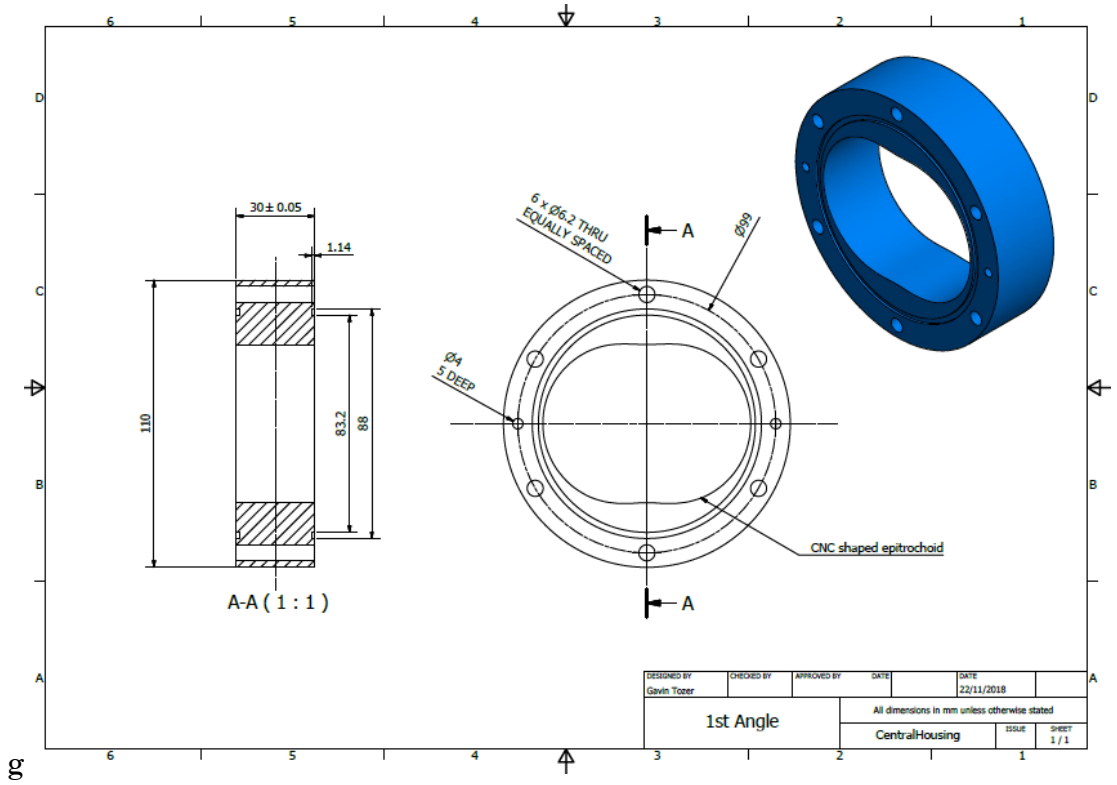


Figure I-1 - CAD drawing: Central housing

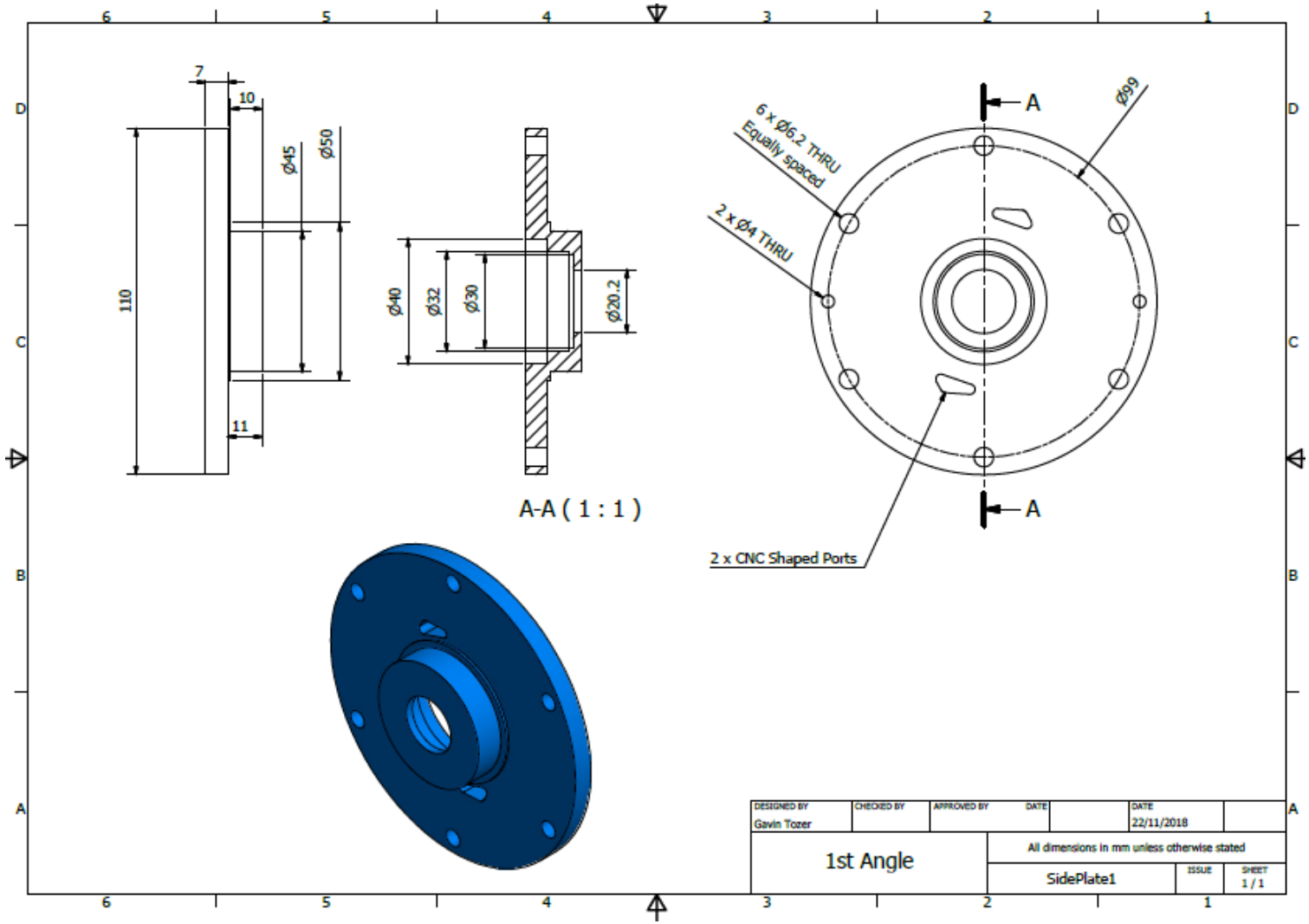


Figure I-2 - CAD drawing: Housing side A

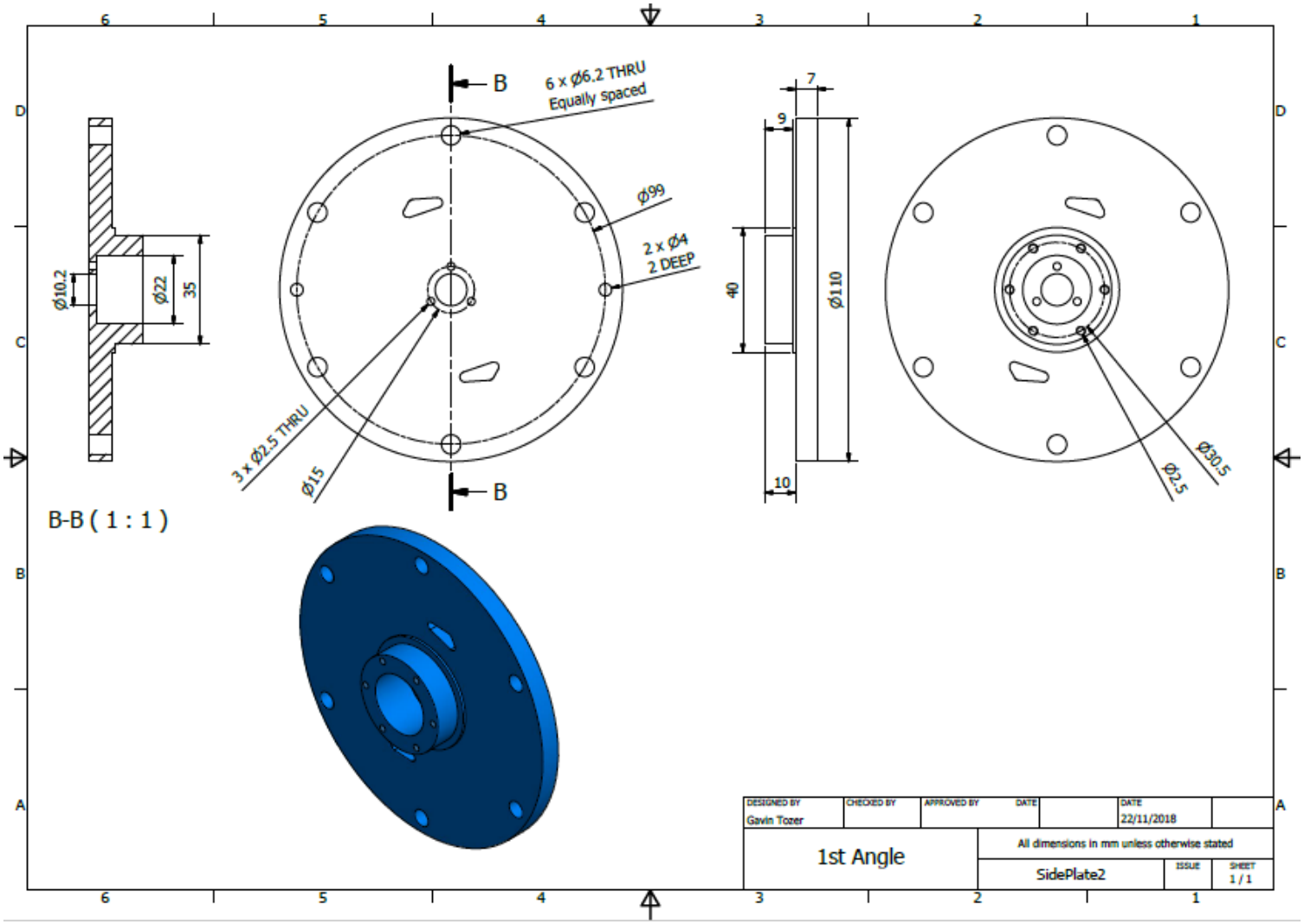


Figure I-3 - CAD drawing: Housing side B

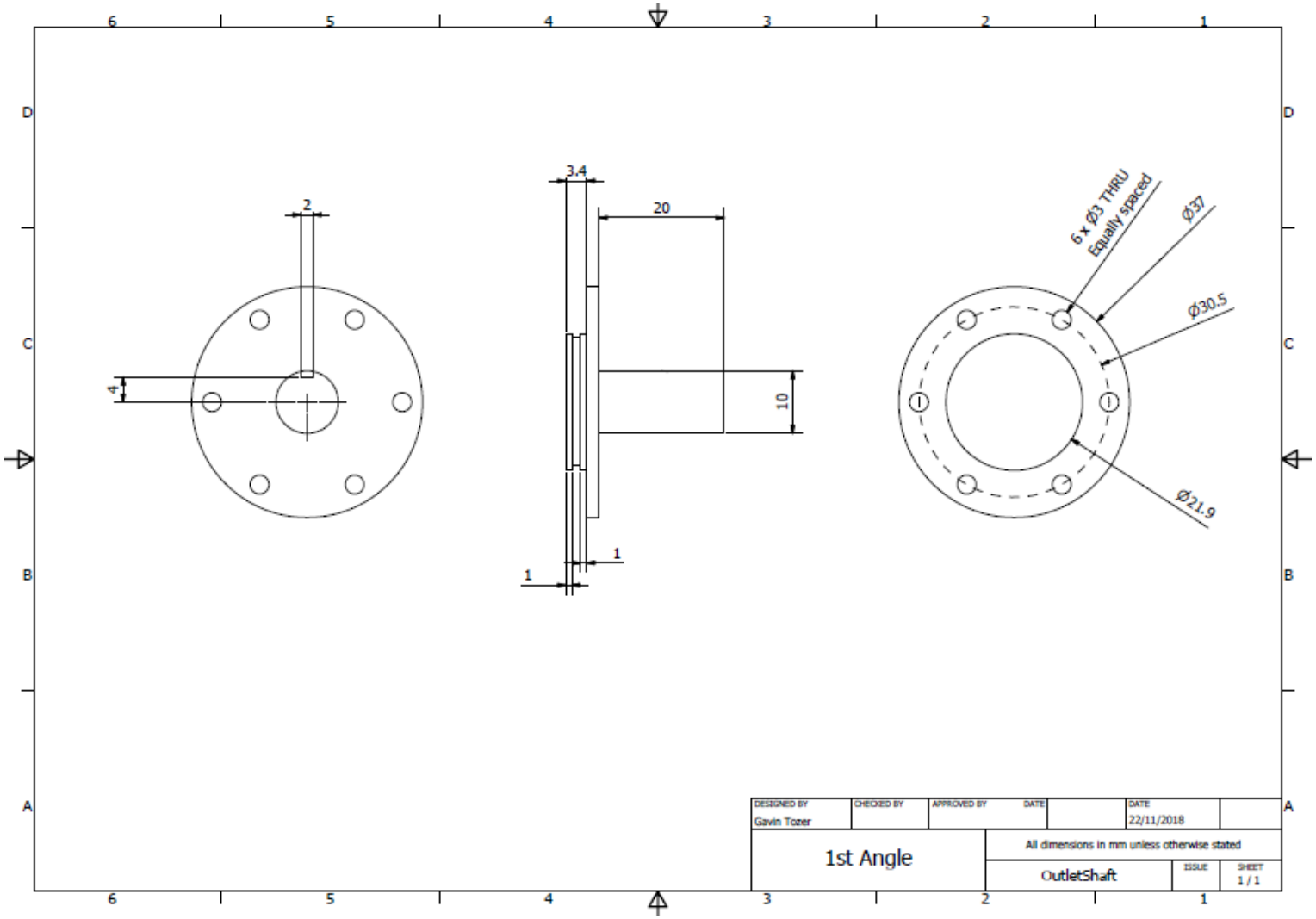


Figure I-4 - CAD drawing: Output shaft

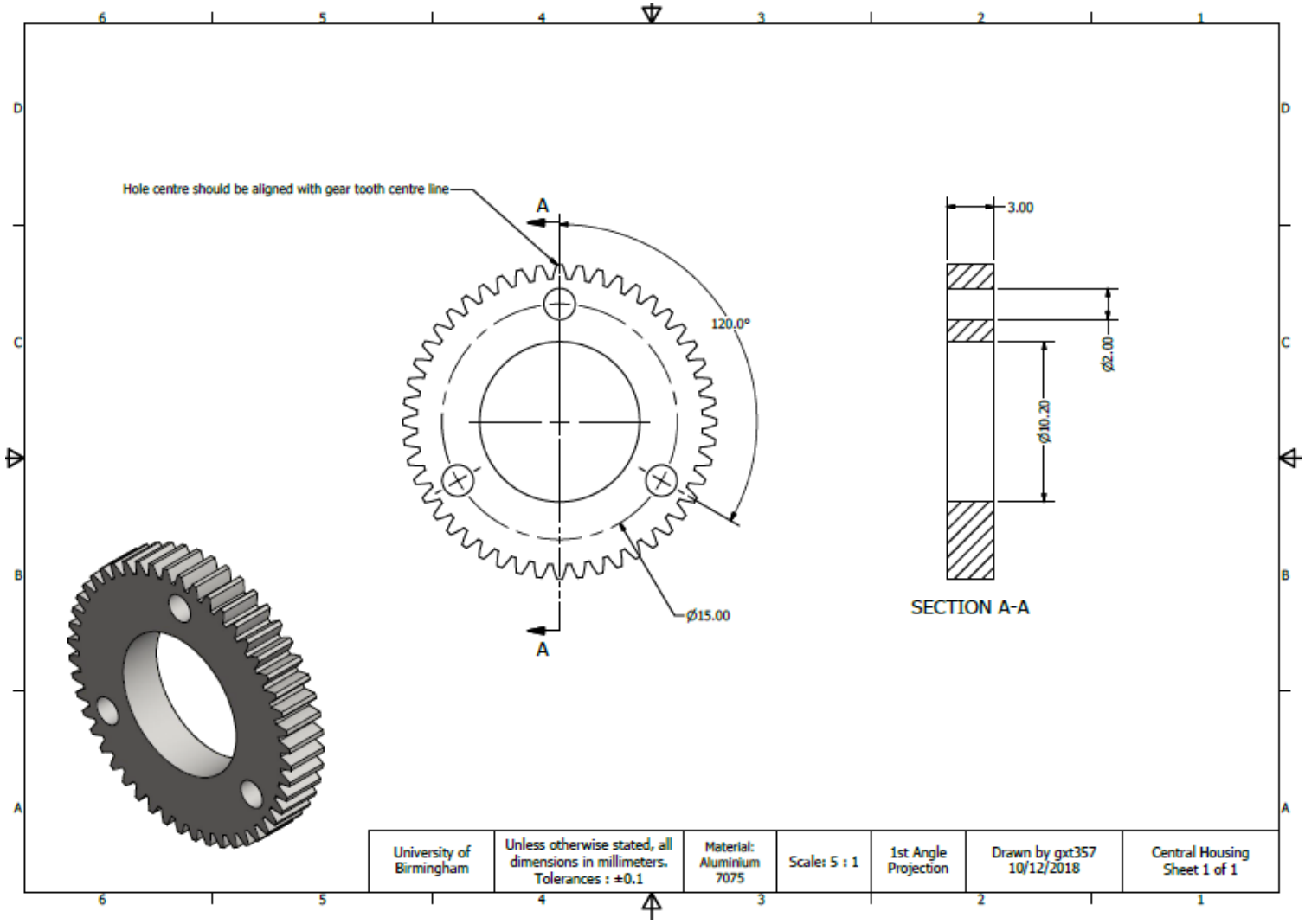


Figure I-5 - CAD drawing: External gear

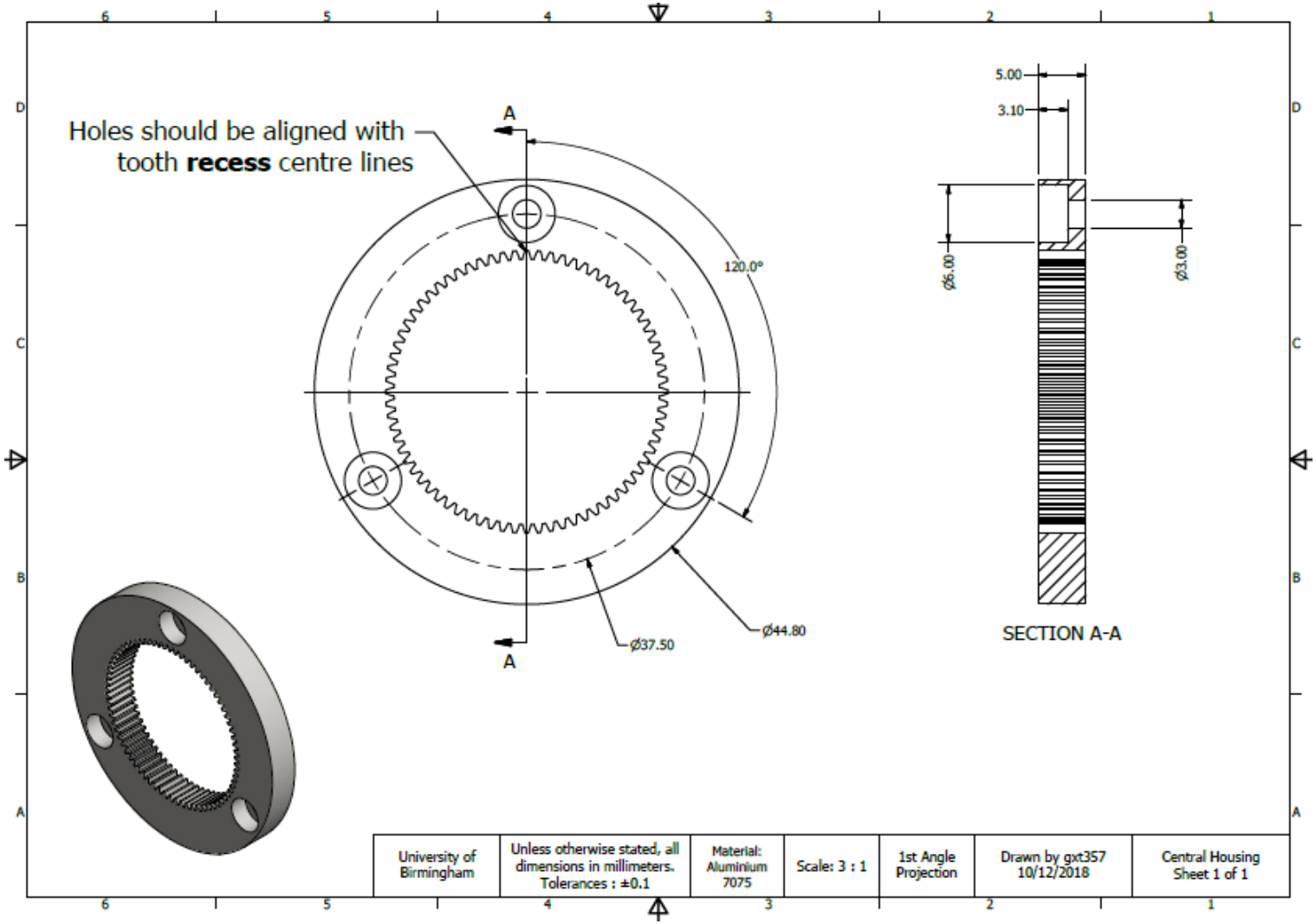


Figure I-6 - CAD drawing: Internal gear

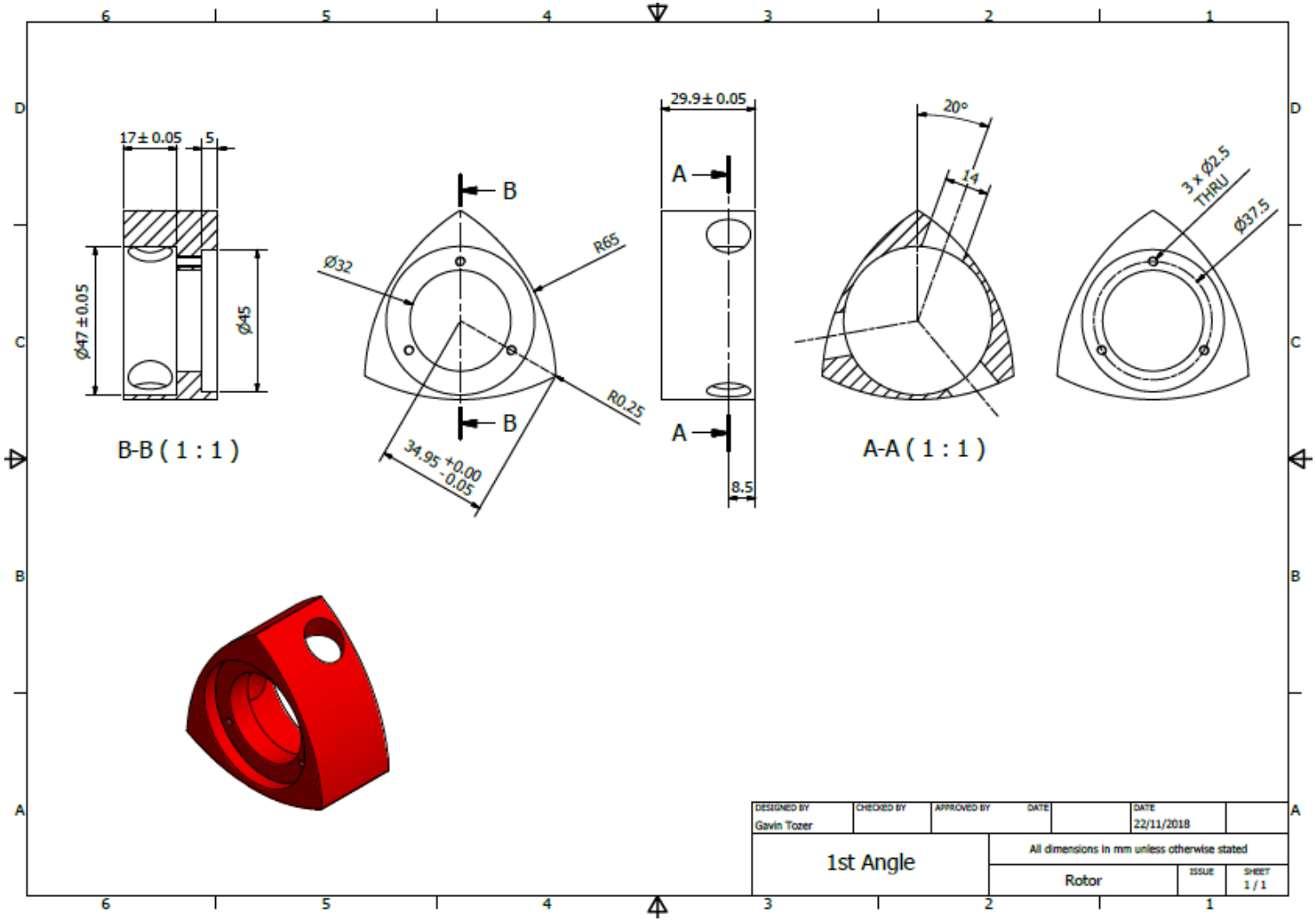


Figure I-7 - CAD drawing: Rotor mk1

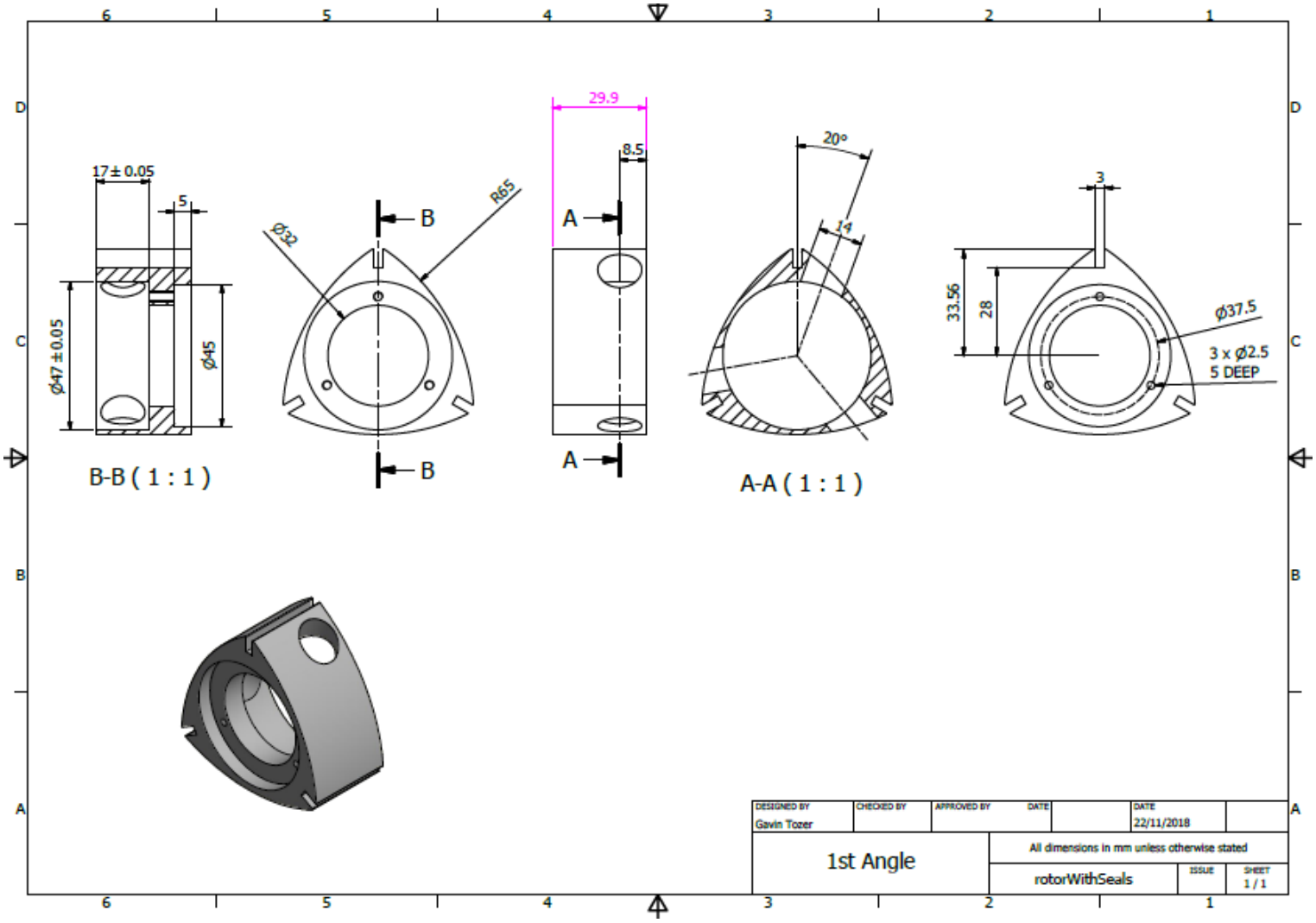


Figure I-8 - CAD drawing: Rotor mk1 (with apex seals)

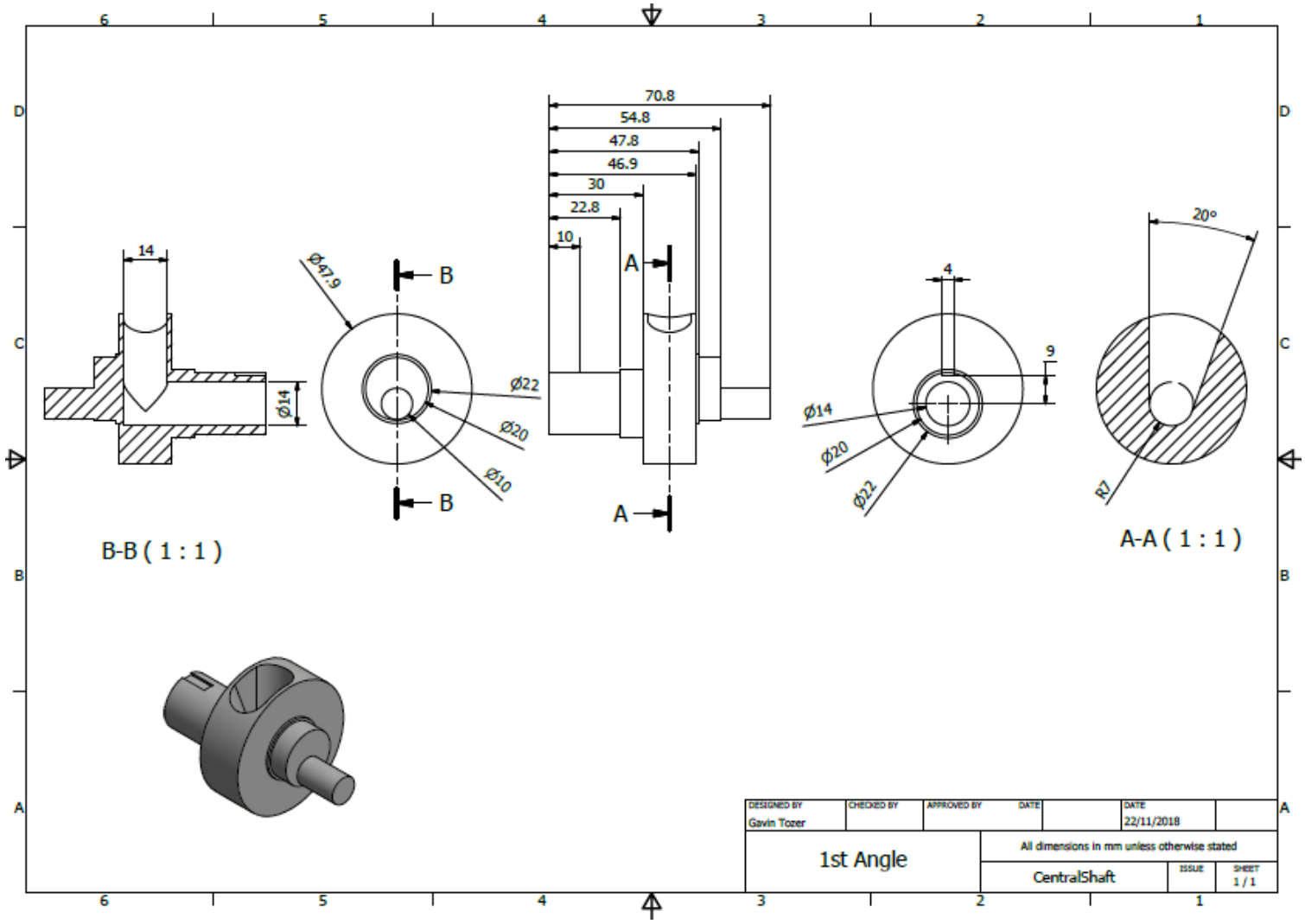


Figure I-9 - CAD drawing: Central shaft mk1

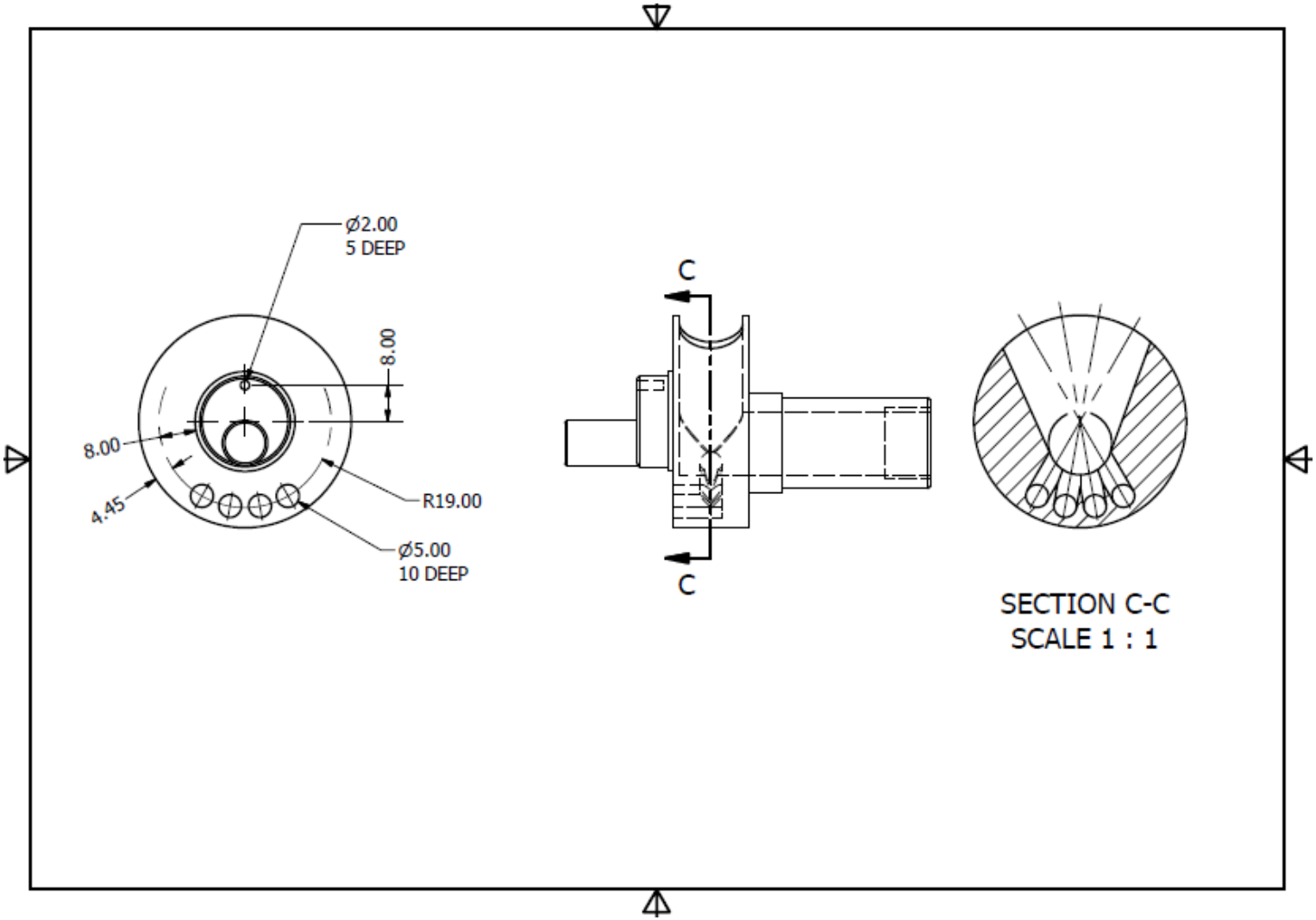


Figure I-10 - CAD drawing: Modified central shaft mk1

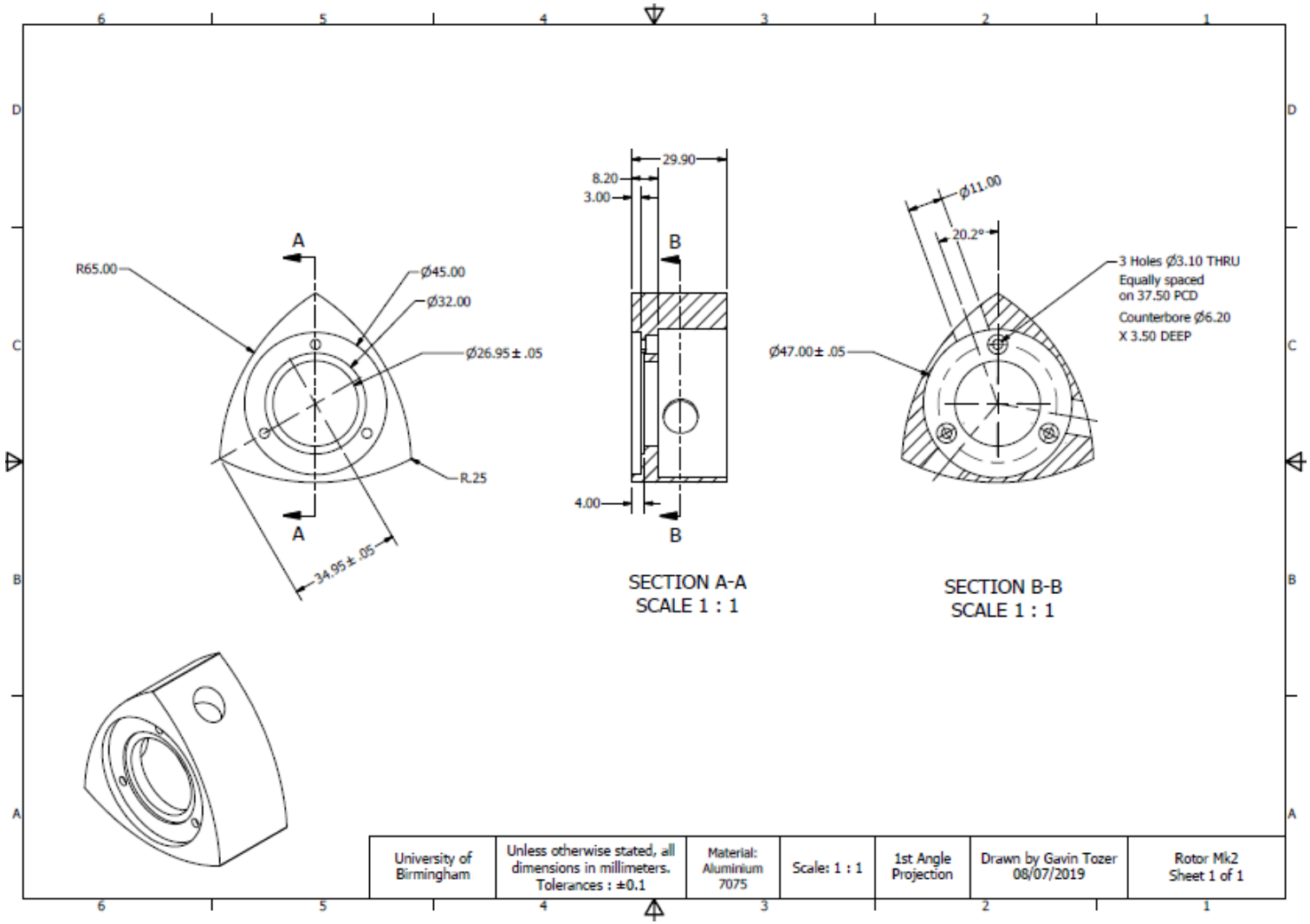


Figure I-11 - CAD drawing: Rotor mk2

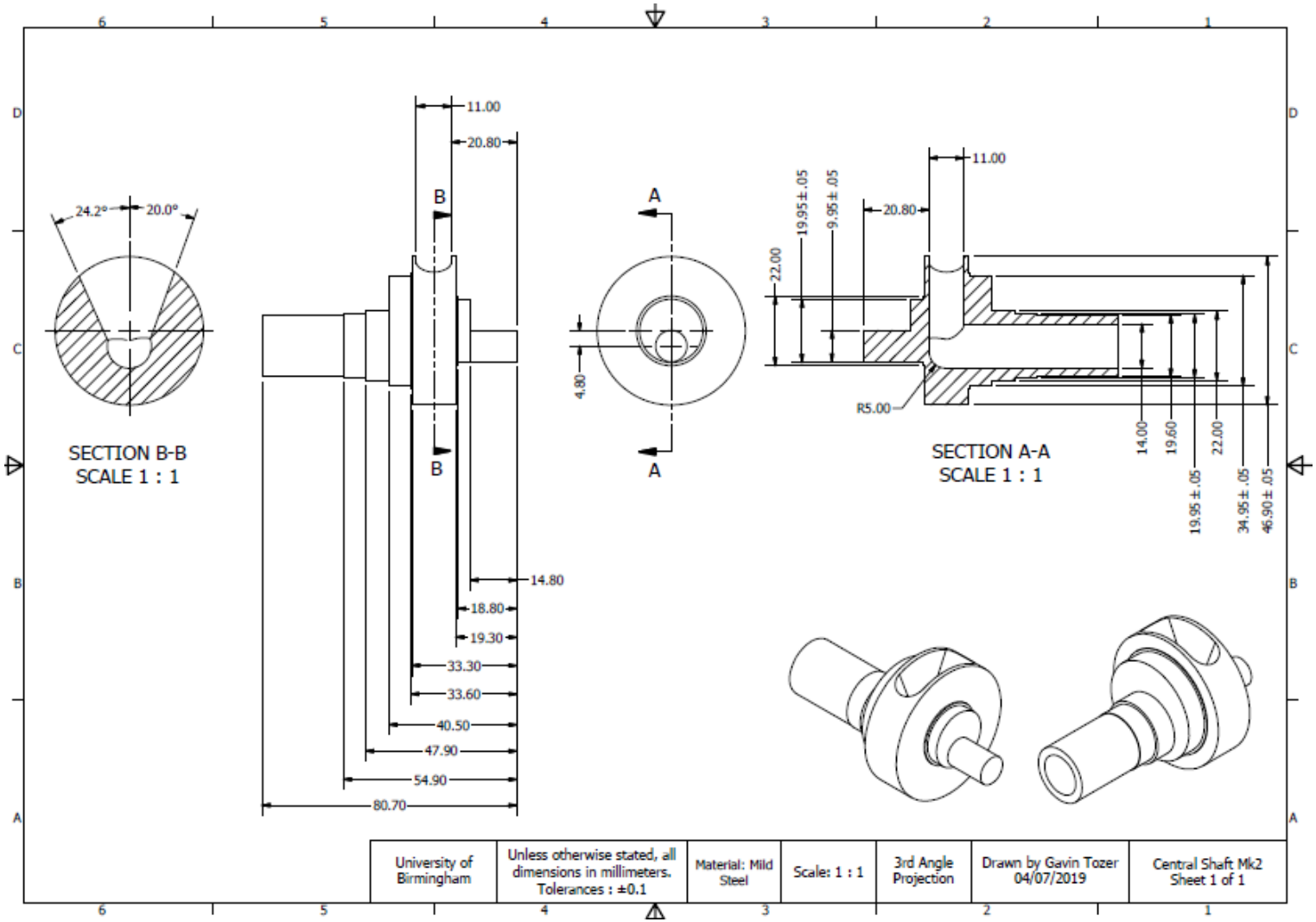


Figure I-12 - CAD drawing: Central shaft mk2