Hybrid Brayton Cycle Model and Facility Commissioning

by

Vitaliy Churilov

A Thesis submitted to the Faculty of Graduate Studies of
The University of Manitoba
in partial fulfilment of the requirements of the degree of

Master of Science

Department of Mechanical Engineering
University of Manitoba
Winnipeg

December 2013

Abstract

There is a lack of available technology to make small-scale power from biomass cost effectively. The proposed Hybrid Brayton cycle is an indirectly heated Brayton cycle with evaporative cooling for combined heat and power generation. It converts a direct fired microturbine to an indirectly heated power system. The Hybrid Brayton cycle offers a flexible biomass power generation platform in the 30 to 250 kW_e range, achieving competitive efficiencies and advantages compared to other systems of similar power level. This cycle is designed to be implemented in remote and off-grids communities, small industries and net-zero communities, where local biomass feedstock is sustainably available. This proposed platform keeps operator qualifications to a minimum. In an effort to validate this new power cycle, a 30 kW_e experimental facility was developed and initial commission phases performed. This facility purpose is to validate numerical model predictions and is used for optimization. The facility is described and results of the commissioning tests are reported with various problems encountered, solutions implemented and recommendations proposed. The thermodynamic model of the Hybrid Brayton cycle is also implemented in the MatLAB environment, incorporating experimental findings and new properties for humidified air at high temperatures. The MatLAB model confirms that an indirect fired Brayton cycle with evaporative cooling could be a viable approach for small scale distributed power generation using biomass. Additional experimental data of humidified air at elevated temperatures would provide more certainty in property predictions. The MatLAB model provides a modeling tool to allow resolving the issues identified during the commissioning of the test facility and offers alternatives to optimize various design configurations, implementing the most up to date property correlations for humidified air at elevated temperatures.

Table of Content

List of Tables

List of Figures

Nomenclature

Acronyms

AC	Alternating Current
CHP	Combined Heat and Power
CR	Compression Ratio
DG	Distributed Generation
ER	Expansion Ratio
ft	Foot
GHG	Green House Gases
GJ	Giga Joule
HBC	Hybrid Brayton Cycle
HE	Heat Exchanger
kW	Kilowatt
lb	Pounds
MFM	Mass Flow Meter
Pr	Prandtl Number
PR	Pressure Regulator
PS	Pressure Sensor
psi	Pounds per Square Inch
Nu	Nusselt number
Re	Reynolds number
Sh	Sherwood number
SP	Burner Sparker
TIT	Turbine Inlet Temperature
TS	Temperature Sensor
UV-S	Ultraviolet Sensor
V	Valve
WDS	Water Delivery System

Notation

A	Surface area over which the heat transfer is occurring
h_s	Heat transfer rate of the source fluid
h_r	Heat transfer rate of the receiver fluid
h_w	Height of the wall
l_w	Length of the wall
m	Mass
P	Pressure
Q	Heat energy
R_T	Total heat resistance of a system
r_w	Heat flow resistance of the wall
r_f	Heat flow resistance due to fouling; fouling factor
$\bar{R}$	Universal gas constant 8.3144 J/mol K
R	Gas constant
Δt	Temperature difference between the source and receiver
T	Temperature
U	Overall coefficient of heat transfer or inverse of total heat resistance
U_T	Overall heat transfer coefficient
v	Specific volume
V	Total volume
x	Molar fraction

Subscript

#	Index number
a	air
e	electric
g	gas
m	mixture
w	water
s	saturation

Acknowledgements

The author wishes to thank several people. I would like to thank my advisor, Dr. Eric Bibeau, for his patience and support throughout my studies in this program and for providing the opportunity for me to work on a new power cycle. I am also extremely grateful for all the support provided by Doug Smith from Entropic Energy. He is the main developer of the power cycle and designed the hardware for the experimental facility I used for this project. Additionally, I acknowledge the support of the Canada Foundation for Innovation and NSERC/Manitoba Hydro Chair in Alterative Energy and the CBIN biomass network. Finally, I would like to thank my fiancée, Serena Vandersteen, for providing much needed love and support during the thesis. Her frequent advice and second set of eyes helped me greatly throughout the research and thesis write-up. I also would like to thank my family for supporting me throughout the duration of the study.

1. Introduction

The global demand for renewable energy solutions is growing steadily. Public interest in regards to current energy availability extends beyond the economic cost of a given energy resource. Many communities worldwide are striving to break their dependence from the usage of fossil fuels for political, environmental and economic reasons. The move towards renewable energy sources promotes reduction of Green House Gas (GHG) emissions and addresses fossil fuels' pick issues. In additional to environmental benefits, renewable energy promotes economic growth, encourages diversity of domestic energy sources, and increases public awareness for better energy policies [1].

The Distributed Generation (DG) concept describes a local power generating setup with reliance on available energy sources. The system may use one or multiple power generation techniques in order to meet local power needs and, if available, provide excess energy to the main power grid. DG provides a platform for alternative power generation for areas, where raw energy concentration is high and its need for energy is sufficiently low, creating economically favourable conditions for an independent power generation system. A range of applications are currently available to support the economic feasibility of DG implementations in communities around the world [2]. Canada generated 12% of its total power in 2003 using DG [3], with anticipated growth in years to come. Although majority of the DG installations currently utilise fossil fuels as main source of energy, the DG system has the capacity to incorporate alternative power generation cycles in order to reduce the overall fossil fuel reliance. The need is present for custom tailored DG solutions that provide efficient and simple power generation systems that meet public needs by optimizing a responsible utilisation of the surrounding local resources.

Biomass is one of the renewable energy sources that is utilised to improve and expand the energy generation industry. Biomass energy is derived from living organisms including but not limited to vegetation, animals and human bio waste. These resources are readily available through sustainable forestry, animal and crop farming, and waste disposal processes within municipalities. With the diffused concentration of biomass in rural areas, where farming, logging and small municipalities are dispersed throughout, there are few engineering solutions

that can economically utilise biomass resources for power generation [4]. Forestry and agricultural economic activities lead to the production of substantial concentration of organic residues. These residues may be detrimental to the environment and the economy of these communities if not properly addressed. Specifically, heavy overgrowth in forests may contribute to more severe forest fires [5], waste from animal farming and processing may lead to water and soil contamination [6], and organic residues can contribution to higher total GHG production when the carbon is not all converted to CO_2 [7, 8] (Figure 1).

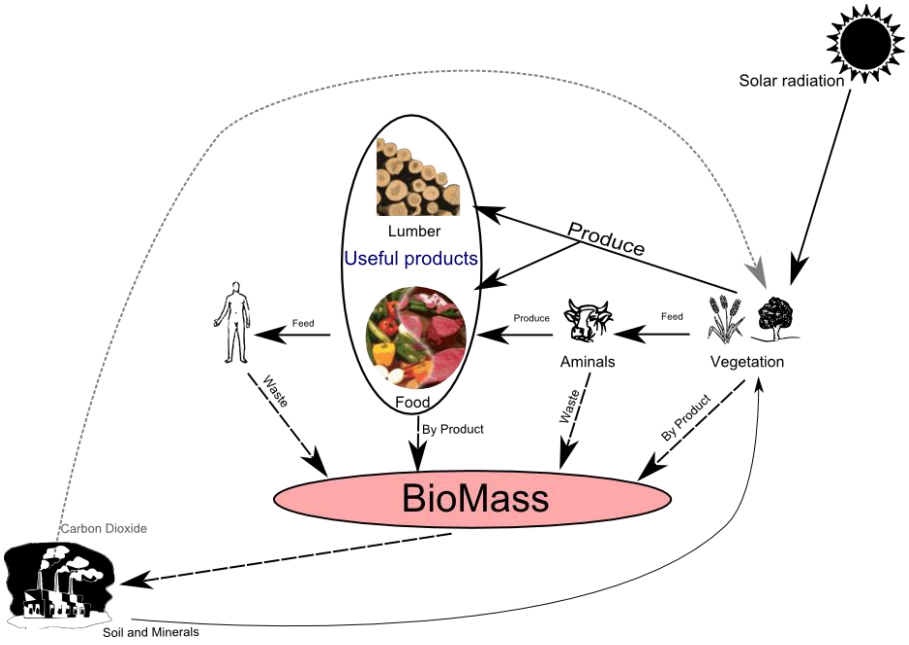


Figure 1 – Schematic of the life cycle of solid biomass products and waste control. The goal is to decompose the biomass to its basic elements with the desire to release CO_2 as a primary exhaust gas back into the atmosphere to be recycled. This creates a closed cycle with living plant biomass absorbing atmospheric CO_2 and releasing O_2. Waste control has exothermic properties that can be used for decentralised power generation. In effect, biomass stores solar energy temporarily using recycled carbon with the energy eventually radiating out to the universe.

All of the aforementioned solid biomass residues can be effectively utilized locally by combustion, thereby eliminating energy and cost for transportation, and the need for large conversion plants that require an economy of scale. It is evident that the potential exists for distributed generation utilising a small biomass power generating unit that can address some of the combined heat and power needs in rural areas. One of the proposed solutions to overcome

the lack of affordable technologies is the newly developed Hybrid Brayton Cycle (HBC), which is described later and is the subject of this thesis.

There are a wide range of potential applications to use biomass residues for sustainable power generation. Biomass can be converted to power, heat or an intermediate fuel. Fuel has unique application for transportation. Some applications focus on developing biomass refinery processes in order to produce biofuel but such approaches require large scale production facilities. However, fuel that used for power or heat generation always produces a reduced overall efficiency as compared to a more direct conversation approach. Alternatively, other strategies try to implement local systems to capitalize on the available biomass in the vicinity. Several approaches can be used for direct conversation of biomass to power and/or heat. It is beneficial to review established thermodynamic principles before reviewing some of the developments in biomass energy conversion.

1.1 Thermodynamic processes overview applicable to biomass power

The piston drive cycles are found in abundance in our daily life and operate mainly using liquid fossil fuels. These systems dominate ground and sea transportation applications, but are also found in remote or emergency power, and Combined Heat and Power (CHP) generation application. Piston engines, in principle, are well equipped to consume biofuels, both in gaseous and liquid forms, and either without engine modification or with some minor adjustments. The International Energy Agency predicts 27% of fuel used for transportation applications will be replaced with biofuels by 2050 [9]. Piston engines are found in CHP generation and consequently can be converted for biofuel applications as well [10]. Conversion of solid biomass to biofuel, liquid or gas and then to power and/or heat is rarely economical due to many energy losses in the overall process including transportation cost of raw material to a central facility, conversion inefficiency to biofuel, subsequent transportation of the fuel to the customer and conversion inefficiency to power and/or heat [11]. That being said, biofuel within the piston engine application has a distinct economic justification for purposes of transportation, emergency and secluded areas' energy needs due to its high energy concentration and ease of short term storage.

One of the most popularly used cycles capable of operating on any fuel source, including solid biomass fuels, is the Rankine cycle; in particular as applied using steam. It has been in use for over 100 years in a range of applications for electric generation, transportation and manufacturing applications. This system is well understood with fully developed engineering and regulatory practices. The Rankine cycle utilises mainly water as a working fluid, under high pressure, usually in closed circuit configuration to actuate and spin a steam turbine which generates mechanical work and residual thermal energy, as outlined in Figure 2 below. Rankine efficiency improvements are directly linked to high pressure and high temperatures piping which is a material challenge and unaffordable for a small-scale system. These difficulties are addressed in medium and large scale steam power systems.

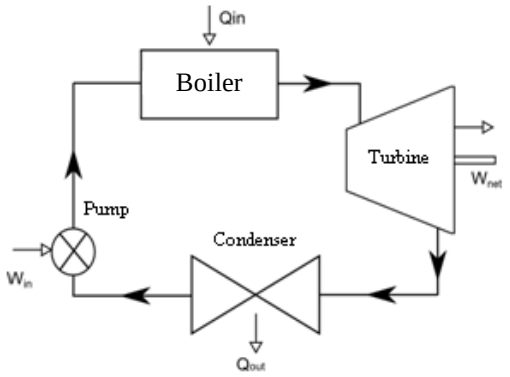


Figure 2 – Schematics of basic Rankin cycle

Steam Rankine applications using solid biomass can be cost effective solutions for medium and large power generation plants above 30 MWe; achieving electric efficiencies of low 30% [11]. Medium and high yield power plants can economically justify additional capital cost to incorporate multi-stage recuperation with a multi-stage turbine configuration and high vacuum condensing to improve the overall plant efficiency. Without these additional systems and limiting temperatures for material considerations, the steam system has a limited capability that will not exceed the 10% – 12% of overall electrical efficiency when using solid biomass in small scale applications [11]. The Rankine system is not economically competitive in small scale applications due to the system's requirement for qualified steam operator(s), often attended legal registration requirements for the steam engineer, large physical plant size, significant initial and maintenance costs and chemistry control of the water. Therefore, users with readily available

biomass generally elect to combust biomass only for heating applications in small scale applications. In North America, steam power size seems require approximately 25 MW to be commercially viable.

The Brayton cycle as shown in Figure 3 is used widely in aerospace applications, in heavy transportation applications (trains, tanks and marine propulsion) and in medium to small size mechanical or electrical power generation. The appeal of the Brayton cycle for industrial purposes results from a high power to weight ratio, higher cycle efficiencies compared to the Rankine cycle, and flexibility of an engine to operate on a wide range of conventional gas and liquid fuels [12].

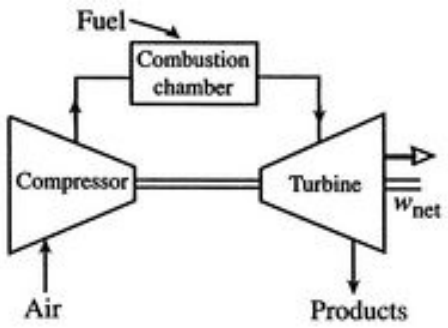


Figure 3 – Basic schematics of typical open Brayton cycle

Microturbines implement the Brayton cycle at a much smaller scale to address specific market opportunities. By imposing a single stage radial compressor and turbine impellers, much lower efficiencies are achieved compared to multiple stage systems. Microturbines add a recuperator to mitigate some of the efficiency losses. The relatively high capital cost of microturbines [13] limits wide application of these systems for electric power or CHP generation using natural gas. Few can economically justify the initial capital investment of the unit for its inherent benefits.

Utilisation of the Brayton cycle with biomass fuel directly fired presents technical difficulties as the biomass must be pressurized, combusted under pressure and its hot flue gas needs to be cleaned to remove abrasive particulates and sticky tars. Alternatively an *indirect* approach is required. Implementation of an indirect approach first limits the total rated power achievable

when using an off the shelf turbine, since approximately 15% of the working gaseous mass is now missing as fuel is no longer injected in the combustion chamber, and lower power production results from indirect heating. Additional differences between the open and closed Brayton cycle that are worth noting is that the energy input into the Brayton cycle changes from being temperature independent in a direct application to temperature dependant in an indirect configuration. Directly fired combustion adds energy to increase the working fluid temperature to peak allowable temperature of the turbine independent of the initial working fluid temperature. However, for the indirectly heated Brayton system, the heat transfer into the system is limited by the temperature of the working fluid entering the flue gas heater exchanger. Recuperation makes the power cycle more efficient by increasing the temperature of the working fluid. However this limits the amount of energy that can be extracted by the hot flue gas since the heat exchanger cannot cool the flue gas below the working fluid's inlet temperature. The effectiveness of the heat exchanger drops as the efficiency of the Brayton cycle increases. The overall efficiency of the indirect Brayton cycle (power generated from energy in the fuel) is the product of the cycle efficiency and the flue gas heat exchanger effectiveness.

The closed Brayton cycle shown in Figure 4 is capable of operating using any heat source. The main difference between the closed Brayton cycle and the Rankine cycle is the fact that the change in the working fluid phase does not occur in the Brayton cycle and therefore the condenser is replaced with a heat exchanger. The working fluid in the closed Brayton cycle is of pure gaseous nature (air, nitrogen, helium) [14, 15] that is usually selected due to its desirable thermodynamic properties. The closed Brayton cycle can be found in medium and large size power generation systems where an abundance of raw thermal energy is available but space is at a premium, thus not favouring a Rankine cycle implementation. Some nuclear powered sea vessels and submarines are equipped with closed Brayton cycles for on-board power generation and propulsion [14, 15].

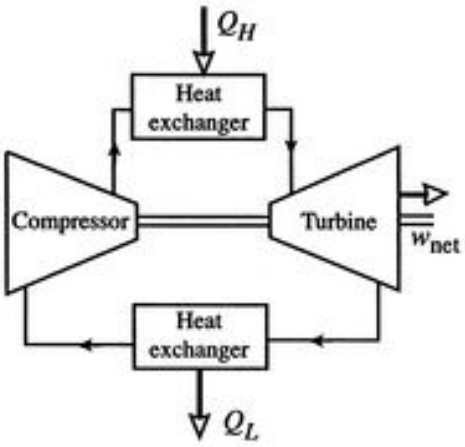


Figure 4 – Schematics of basic configurations of closed Brayton cycle

1.2 Systems under research and development

There are number of designs on the market that attempt to utilise and/or improve upon the fundamentals of the basic Otto (piston), Diesel (piston), Rankine (steam) and Brayton (jet) cycles in an effort to integrate biomass applications. The successes and challenges of these applications are summarised in this section in an effort to outline the distinctions between these systems and the new HBC system being studied. Table 1 outlines the characteristics of each system at the small scale level with efficiencies obtained though modeling of the cycle and all auxiliary processes, including fuel preparation if required [11].

Currently, conversion of biomass into biofuel can be achieved via physical (e.g. pellets), chemical (e.g. Fischer–Tropsch process), biochemical (e.g. fermentation) and thermo-chemical processes (e.g. pyrolysis or gasification) [11]. The successes of these processes range from experimental systems to production plants but market penetration remains limited. Refined biofuels used for electric power generation will be restricted to single digit efficiency from biomass to electricity production [11]. The inefficiencies of the refining process and inherent losses of the power plant make this process only economical in applications where high energy density fuels are desired, like emergency and secluded area power generation and transportation. The incorporation of refined biofuel products in Canada is seen today for transportation applications within the existing infrastructure, 5% of ethanol and 2% of bio diesel is blended in to complement transportation fuel needs [16].

There are numerous attempts to gasify biomass, clean the biogas, and inject the biogas directly into a piston engine. There are many impediments and these systems have relatively low efficiencies; below 10% in small scale. The literature reveals gasification systems quoting much higher efficiencies however scrutiny reveals that these are efficiencies of subcomponents of overall systems. In small scale when accounting for all conversion steps, the reality is that such systems have a very low overall efficiency compared to other approaches [11]. The advantage of using a low cost engine must be outweighed with the technical difficulties of flue gas cleaning, high maintenance costs, and low overall efficiencies.

The availability of the steam engine in the small scale range is limited and can be obtained as a customized solution from vendors.

Improved designs of the conventional Rankine cycle for small scale can be seen on the market in the form of the Organic Rankine Cycle (ORC) with the working fluid replaced by an organic fluid or synthetic oil. The system capitalises on the properties of selected organic oils that operates in a different temperature/pressure regime than water and is more amendable to recuperation due to the shape of the saturation line bending inwards. Higher pressure vapour exiting the turbine can reduce the turbine impeller size compared to an equivalent steam system [11]. The idea of introducing synthetic fluid of greater density than water partially compensates for the lower enthalpy of non-water fluids. The ORC has been a known technology for over fifty years and has been somewhat successfully applied in Europe in small biomass CHP systems. The system is complex, is typically designed to use two separate fluid loops in series and is designed to circumvent high pressure boiler regulations [11]. Consequently, ORC design results in relatively high initial capital cost and some losses in efficiency. The ORC improves the Rankine performance and delivers 10% – 13% biomass fuel to electric efficiency [11, 20] and is commercially available in a configuration as small as 400 kWe from Turboden [17].

The cogeneration Entropic cycle is a new development design that expands the Rankine cycle beyond the ORC. The patented Entropic cycle uses a water/ammonia mixture working fluid that recuperates latent heat within the cycle, does not require a secondary fluid loop for the flue gas

heater and can reject heat at temperatures conducive to CHP. It is anticipated to deliver 14% electric efficiency in its CHP operation mode which can be increased if lower reject temperatures are designed. The patented system is under development by Entropic Energy [11, 18].

There are several design configurations where the Brayton cycle has the potential to use biofuel applications outside the conventional gas turbine application which has been widely in use since World War II. One such design is the evaporative gas turbine cycle (EvGT), which utilises a process of water addition to the air mass flow in the humidifier, after the compression stage. This is done in order to reduce the temperature of the air and to increase the mass flow rate across the turbine. With the recovery and reintroduction of energy through water vapor into working fluid, the natural gas fired EvGT system is expected to have an electric efficiency of 45% to 50% [19]. A first of its kind pilot plant of EvGT 600 kWe was operational since 1998 at Lund University [19].

An indirect gas turbine utilises a heat exchanger to heat up compressed working fluid gas instead of combusting a fuel within the working fluid in a combustion chamber. The indirect fired gas turbine, also known as an air turbine, has operationally different limiting factors from the conventional gas turbine arrangement. In conventional gas turbines, a higher Compression Ratio (CR) yields improvements in efficiency. This is due to the fact that when air enters the combustion chamber at a higher pressure and temperature, it requires less fuel to reach the max Turbine Inlet Temperature (TIT), which is limited by the material properties of the turbine. Air turbine design, on the other hand, needs to balance the CR with the operational temperature of the heat exchanger [11]. An increase in CR results in an increase in pressure and temperature of the air, which consequently reduces the amount of energy transferred from the flue gas heat exchanger. The optimal balance is usually achieved at 7 – 10 bar which yields a 7% – 8% electric efficiency of the system [11].

The systems discussed above are designed in an attempt to address the need for small scale biopower CHP systems. The simplicity of the system and the initial capital investment are the driving forces for development for this market [20]. The direct injection Brayton cycle, like EvGT 600 [19], focuses on modifying the direct fire application, due to the potential system's

high efficiency yield [21]. This approach will fail to produce a commercial system due to fouling and deposition on the turbine blades. The indirect Brayton systems that are currently in development for applications of biomass do not attempt to control the temperature via water injection [21, 22]. In contrast, these systems in development attempt to increase the system's tolerance for higher temperatures by incorporating ceramic heat exchangers and heat treated turbine blades [22, 23]. However, higher temperatures increase the Brayton cycle efficiency at the expense of the heat transfer into the cycle causing a trade-off effect which mitigates benefits of higher cycle efficiency on overall efficiency. Moreover, communities require the ability to maintain their own equipment without relying on specializing trades. The Hybrid Brayton cycle makes indirect firing possible at relatively low temperatures to allow using common materials by capitalising on the temperature and mass flow effects of water injection. Water injection allows for improved working fluid temperature control throughout the process, with minimal effects on the cycle efficiency and great improvement to overall system efficiency.

Table 1 – Alternative fuel systems comparison chart
with theoretical values added for Hybrid Brayton cycle [20]

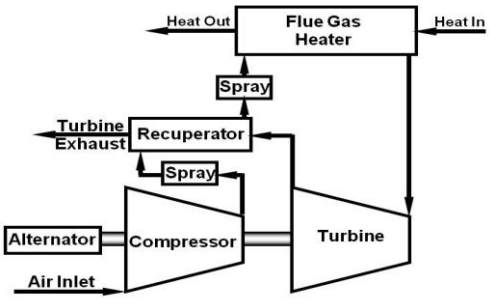
System Comparison	Large Steam Cond. Cycle (Williams Lake Power)	Bio-oil Conversion (Pyrolysis)	Gasification Conversion (Syngas)	Biofuel cycles equpied with combuster or gasifiers					Hybrid Brayton Cycle (Theoretical Values)
				Brayton Cycle	Small Steam Cond. Cycle	Small Steam CHP Cycle	Turboden ORC Cycle	Entropic Cycle	
Technical Complexity/Operator Qualification low 1.....5 high	4/5	5/3	3/3	1/1	4/4	3/4	3/2	2/2	1/1
Equipment Size and count less1.....5 more	5	4	3	5	5	4	2	1	2
Fuel Preparation less1.....5 more	2	5	4	2	2	2	2	2	2
Capital Cost($/kW) low 1.....5 high	2	5	3	4	5	3	4	2	3
Optimum System Size	60MW	>5MW	<2MW	<1/2MW	<5MW	<5MW	<2MW	<2MW	<1/4MW
Power Efficiency - Biomass to electricity	29.2%	6.4%	7.8%	7.4%	9.9%	5.7%	10.2%	12.0%	~12.0%
Secondary Products yes 1.....5 no	5	4 (Char)	5	5	5	3	2	1	5
Overall Cogeneration	29.2%	6.4%	7.8%	7.4%	9.9%	53.9%	54.5%	67.7%	~55%

1.3 Hybrid Brayton cycle overview

The goal of the HBC design is to expand the current biomass power generation spectrum to include cost-effective small scale CHP, taking advantage of the development of microturbines. The system needs to be simple, flexible, affordable and compact. HBC design incorporates some of the air turbine principles and adds advantages from air humidification and the Rankine cycle. The HBC design introduces controlled flow of water injected into the compressed working fluid before and after the recuperator in order to reduce the working fluid's temperature prior to entering heat exchangers. This practise increases the amount of mass passed through the turbine, an important aspect when using an indirect fired microturbine that was designed to be direct fired with natural gas. Furthermore, the working fluid exhaust exiting the turbine produces low pressure clean moist air at approximately 325°C, which can be adapted for space heating applications.

Another HBC goal is to deliver a small CHP system that operates on any biofuel, biomass and conventional fuel sources without requiring modifications to the power plant. The Hybrid Brayton cycle's name is indicative of it being a combination of existing techniques. It is essentially an open Brayton cycle indirectly fired that has adopted some of the Rankine cycle characteristics, like indirect heating and use of water as a working fluid pressurized by a liquid pump rather than by a compressor. The HBC design uses an off the shelf microturbine and a new system controller to provide required flexibility to the wide range of user needs, as shown in Figure 5.

Figure 5 – Schematic of typical Hybrid Brayton cycle configuration with independent heating system feeding a flue gas heater and a recuperator, as envisioned for manufacturing purposes [24]



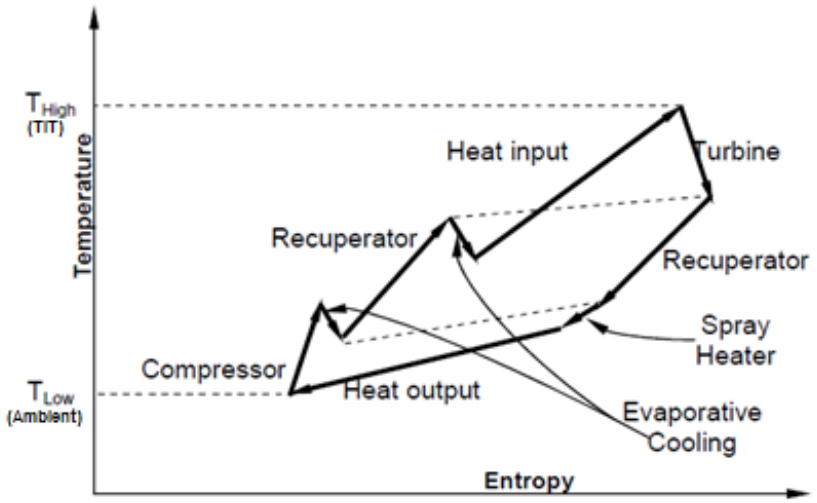


Figure 6 – TS diagram of HBC cycle with spray before and after the recuperator. Note that the overall mass flow changes throughout the cycle, unlike traditional TS diagrams [24].

Figure 6 shows the TS diagram of the more complete implementation of the of the HBC cycle with water injections before and after the recuperator. The first injection after the compressor, before the recuperator, uses evaporative cooling to reduce the air temperature, increase the entropy and increase the mass flow of the working fluid. Reducing the temperature of the working fluid at this location increases the heat recuperation. The added mass increases the power produced by the turbine. The amount of evaporative fluid that can be added is related to the saturation capability of the air at the local P and T. This added water mass increases the turbine power generated without increasing compressor parasitic power, a critical aspect for this indirect Brayton cycle. Evaporative cooling can also be applied after the recuperator. A higher temperature of the humidified air after the recuperator allows for more sub-saturated injection of water. The decrease in the average temperature of heat input reduces the cycle efficiency but this decrease is more than compensated by the added mass and increased energy into the cycle which increases the power produced by the turbine. The net effect is only a slight decrease in cycle efficiency due to the evaporative cooling following the recuperator but a large increase in system efficiency. The HBC achieves the full power rating of the microturbine while allowing for more effective control strategies by regulating how much fluid is injected after the recuperator compared to before (defined as the water split ratio injection).

From simulations, it is found that up to six times more water compared to the fuel can be injected to bring the air to saturation. The cycle overall efficiency can be further improved by heating the injected water with the exhaust flue and adding an additional spray before heat rejection in a closed version of the cycle, as shown in Figure 6.

1.4 Thesis overview

Implementation of a design from the form of an idea to its production requires many steps of research and development. This thesis addresses some of these steps in an effort to answer key questions of performance, design requirements, and limitations for the HBC model. This thesis demonstrates the thermodynamic potential of HBC, by developing a transparent simulation and modeling tool. The experimental setup addresses practical questions of power plant configuration in terms serviceability and functionality.

1.5 Modeling

The cycle performance is dependent on the humidified air properties at elevated temperatures, an application which in engineering is relatively rare. Determining thermodynamic properties at the temperatures and pressures of this power cycle must be accomplished using a suitable algorithm as experimentally confirmed tabulated values are not available. Thus this thesis examines three distinct simulation approaches for the humid air properties that are intended to simulate the thermodynamic performance of the HBC. By examining these three approaches, the thesis underlines the performance of each thermodynamic property model in an effort to identify the most suited approach for modeling the thermodynamic process of the HBC cycle and establish a conservative prediction of cycle efficiency. This thesis also aims to provide an optimisation numerical tool to improve the HBC design in future studies, as detailed in Chapter 5.

1.6 Prototype

Construction of a prototype R&D system is resource intensive and a time consuming process. The benefits of such a practise, however, are invaluable to the effort of developing a commercial

unit. This thesis will outline the HBC prototype setup and data acquisition instrumentation layout. There is also included documentation of many lessons learned and design recommendations based on observations obtained throughout the construction and commissioning process. For example, Chapter 3 discusses the setup of the HBC prototype and Chapter 4 outlines a variety of recommendations on procedures that were developed as a result of many attempts to start the system in a safe manner.

1.7 Contributions

The contribution of this work related to the assembly of the HBC's experimental facility and its commissioning, and the design and implementation of the data acquisition system and instrumentation. This also includes the documentation of the HBC's instruments, power generation and other utilities. In addition, a thermodynamic model is implemented in MatLAB. More specifically, contributions relate to developing the instrumentation for the laboratory system, helping in assembling the facility, developing a start-up/shutdown safety manual, designing and debugging the data acquisition and processing software. In conjunction with commissioning efforts, three distinct HBC simulation models are developed in the MatLAB environment to analyze the HBC system using the numerical model. Simulation models are tested for thermodynamic predictability while simulating HBC system and optimal model is selected.

2. Thermodynamic properties of air/water mixture at high temperatures

It is beneficial to research past modeling studies for air/water mixtures as this will provide a frame of reference for the validation of the HBC model and an indication of HBC's thermodynamic performance. In order to model the HBC with accuracy, the model approach utilises thermodynamic property models for air, water and its mixture.

2.1 Indirect fired Brayton Cycles with water injection

The HBC design partially addresses the thermodynamic limitation of the externally fired Brayton cycle. The water injection into two pressurized sections of the HBC is a critical step to improve efficiency. Horn in his PhD dissertation [25] outlined the development of the introduction of water into gas turbines. Horn delineated studies which had shown that the introduction of water into a gas turbine is beneficial in order to increase thermodynamic and cycle efficiencies for direct fired turbines. Reduction of working fluid temperature during the compression phase results in an increase of compressor and combustor efficiencies in a Brayton cycle [25]. The process is well understood and is used in industrial power generation applications like General Electric LM6000 SPRINT gas turbine power plant [25]. Horn et al. focused their studies to understand water effect on conventional Brayton cycle. There is no study or publication other than the HBC patent that attempts to understand the effects of water introduction into working fluid in an indirect Brayton cycle, the approach used in the HBC before and after the recuperator. Furthermore, a patent search reveals that no such device has been patented prior. The HBC is patent pending to cover the main cycle and control strategies when coupling the cycle to a combustion system or a solar dish, as the water split ratio and injection can produce benefits not discussed in the open literature.

2.2 Thermodynamic simulation

The concept of the Hybrid Brayton cycle was developed mainly by Doug Smith from Entropic Energy and remains largely unpublished. Smith and Bibeau have been working on this cycle since its creation in 2005. Smith initially developed and improved on the idea of HBC using a

modified algorithm extrapolated from the Nelson and Sauer study [28] to demonstrate a theoretical viability of the cycle. The model outlined the potential feasibility and practical potential of the HBC in terms of cost, efficiency and physical space. Smith also developed a simplified HBC design for the laboratory system assembled and commissioned in this study at the University of Manitoba.

No studies can be found in the area of indirect Brayton cycle with water injection and as such, this thesis draws on core thermodynamic models available in order to simulate the behaviour of air and water in an assumed operational range of high temperatures and relatively low pressures. The thermodynamic characteristic of dry air, pure water and their proportional presence in mixed fluid derived from experimental data and presented in empirical notation. As part of the European Advanced Adiabatic Compressed Air Energy Storage (AA-CAES) project, Herrmann et al. conducted a comprehensive study of available thermodynamic humid air property models as compared to the available experimental data in an effort to identify the optimal and comprehensive humid air model [26]. In order to achieve that objective, Herrmann et al. conducted experimental validation of the models to fill in some of the gaps that existed prior in range of published experimental data for properties of humid air [26]. Their study, published in 2009, concluded that modified virial equation of state by Nelson and Sauer [27] is optimal to determine thermodynamic properties, and for transient properties an improved Vesovic-Wakeham model was developed [26]. Their proposed thermodynamic model can be used to predict the thermodynamic behavior of humid air at a wide range of temperatures (243 K – 2000 K) and pressures (611.2 Pa – 100.0 MPa) with minor restrictions associated with the calculation of transport properties [26].

Although the AA-CAES study utilises some of Nelson and Sauer model [27], the original HBC design and proof of the concept was based on the later model. It is worthwhile to review and understand the Nelson and Sauer's model [27] and compare it to Herrmann et al. [26] findings in an attempt to develop a more accurate HBC simulation, as this will impact the overall system efficiencies. Nelson and Sauer's paper offers a virial equation of state that provides thermodynamic properties of moist air humidity ration for 0 to 1 kg_w / kg_a, temperatures -40°C to 320°C and pressures from 0.069680 MPa to 5 MPa [27]. Both the humidity ratio and pressure

range satisfy the requirements for the HBC simulation. Nelson and Sauer acknowledge that the coefficients used in their equations can only be reliable for temperatures up to 320°C [27]. At higher temperatures the errors from coefficients exceeded 2% and they stop their validation above 380°C [27]. Consequently, the temperature range from 320°C to 735°C has an unknown error that can exceed 2% error inherited in the original HBC model.

This thesis recognises this limitation of the original model and attempts to rectify it by developing a more valid modeling approach for the full range of HBC operation, up to 735°C (1,350^{o}F). In the simulation of humid air, fundamental equations of state are usually utilised to predict the fluid properties of the fluid. These equations are fitted with empirical coefficients extrapolated from experimental data. Nelson and Sauer utilise the virial equation of state, originally developed by Rabinovich and Beketov [28], for their proposed humid air model [27] generating a real mixture of real gases model. Alternatively, Herrmann et al. selected to build their recommended model using the Helmholtz energy function [26], a simulation of ideal mixture of real gases [26]. The real mixture of real gases model considers the mixing effects of the substances on each other. Each model utilises existing resources to determine separately the thermodynamic properties of dry air, water (ice, liquid and gas) and the composition of the saturated water within the air sample in order to predict the thermodynamic properties of the humid mixture. HBC simulation utilises water properties of liquid and vapor states throughout this study.

Nelson and Sauer use the National Institute of Standards and Technology (NIST) thermodynamic properties of dry Air [29] which is valid for up to 2000 K and 70 MPa. The dry air properties are given in terms of Helmholtz function but Nelson and Sauer adjusted the equations to work with equations of state and virial coefficients through curve fitting [27]. Herrmann et al. [26] proposes in his recommended model to utilise the same properties and maintains them in Helmholtz configuration. Nelson and Sauer [27] then selected the most up to date thermodynamic properties for water and steam from International Association for the Properties of Water and Steam (IAPWS) [22]. In 1997, IAPWS updated the formulation for the application of water and steam in industry in order to increase the accuracy and calculation speed of the thermodynamic properties which are derivatives from Gibbs function [27]. Herrmann et

al.utilises the IAPWS-95 as a recommended method for thermodynamic properties of water and stream which are provided in Helmholtz energy notation [26]. For the composition of the saturated fluids, Herrmann et al.utilise the Nelson and Sauer method for determining the saturation composition, but updated the second virial constant for water-water (B_{ww}) and air-water (B_{aw}) [26].

Each model that is used for the HBC simulation has an inherent list of assumptions within its mathematical limitations [26, 27]. These assumptions are applied during the HBC simulation as well. Additional assumptions were made during the write up of the code and are annotated in the script itself.

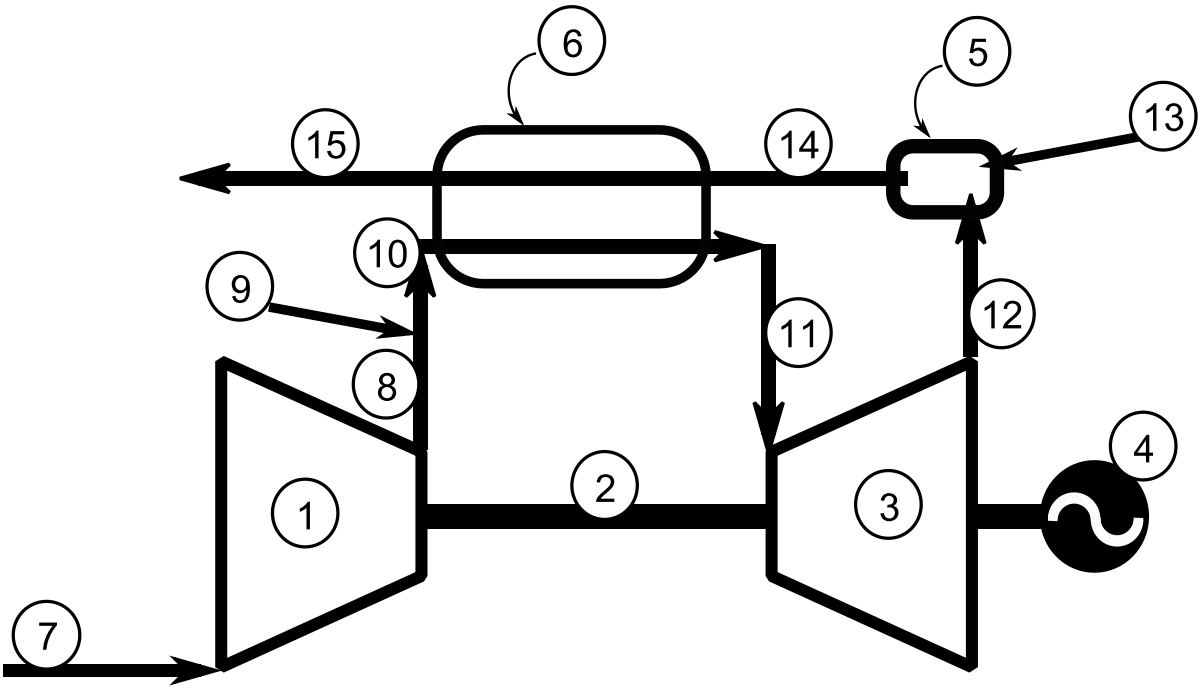
It is evident that the goal of AA-CAES project was to develop a model that has the capacity to generate thermodynamic characteristics of humid air in a wide range of temperatures and pressures. To obtain that range and maintain sufficient accuracy, Herrmann et al.recommended in their study an ideal mixing of real gasses as a model of choice [26]. They also used a Helmholtz energy notation for their model development, fundamentally different from Gibbs function used by Nelson and Sauer [27]. The Helmholtz energy notation bases it thermodynamic calculations on mixture's properties of density and temperature variables [26] but Gibbs function sets pressure and temperature as variables of choice [27]. The study predicts that Helmholtz energy notation will be more difficult to simulate computationally as initially the HBC simulation is set in terms of pressure and temperature inputs.

A summary of Nelson and Sauer equations that are used in simulation models mentioned above can be found in Appendix A. In addition, a printout of simulations' Mathlab script can be found in Appendix C for both mixture models.

3. Experimental HBC test facility

The simplified experimental HBC laboratory design is outline in Figure 7 that contains a single spray injection. The figure presents all major components of the test facility. The figure also identifies critical stages where the thermodynamic state of the working fluid is measured to determine the overall performance of the cycle. The laboratory system does not implement recuperation so only one water injection is used. This differs from the actual design and patent which achieves a higher efficiency and more controls when implemented with combustor or concentrated solar applications. The laboratory system also introduces fuel for energy input rather than biomass flue gas for simplicity and controllability reasons.

The HBC intakes atmospheric air at ambient temperature and passes it through a single stage compressor where the temperature and pressure of the working fluid increases. Water at 150 psi pressure and room temperature is injected into the working fluid via direct injection, resulting in a reduction of working fluid temperature and increase of the working fluid mass flow. The heat exchanger has a counterflow design that allows the working fluid to gain energy. The temperature of the working fluid will rise to turbine inlet temperature (TIT). The turbine will produce work by expansion of the working fluid through its single stage turbine blades. The turbine utilises direct drive to operate the compressor and to produce electric power. The working fluid leaving the turbine has lost most of its pressure and some of its temperature at this point. Propane is injected into a high flow rate burner to reheat the working fluid which results in an increase of the working fluid temperature (and enthalpy) to the max temperature design. After the burner, the working fluid passes though the heat exchanger, where it loses the energy to the clean working fluid and the exhaust to the atmosphere. The HBC prototype is designed by Entropic Energy utilising T304SS, and the installation is shown in Figure 8 to 10.



Component	Stages
1. Compressor	7. Intake Air
2. Shaft	8. Compressed Air
3. Turbine	9. Water
4. Electric Generator	10. Compressed Air with Water
5. Combustor chamber	11. Heated Up working Fluid
6. Heat Exchanger	12. Working Fluid past Turbine
	13. Fuel
	14. Heated work fluid after fuel combustion
	15. Exhaust

Figure 7 – Schematics of the layout of the Hybrid Brayton cycle experimental system setup with indexed description of its components and functional areas as built.

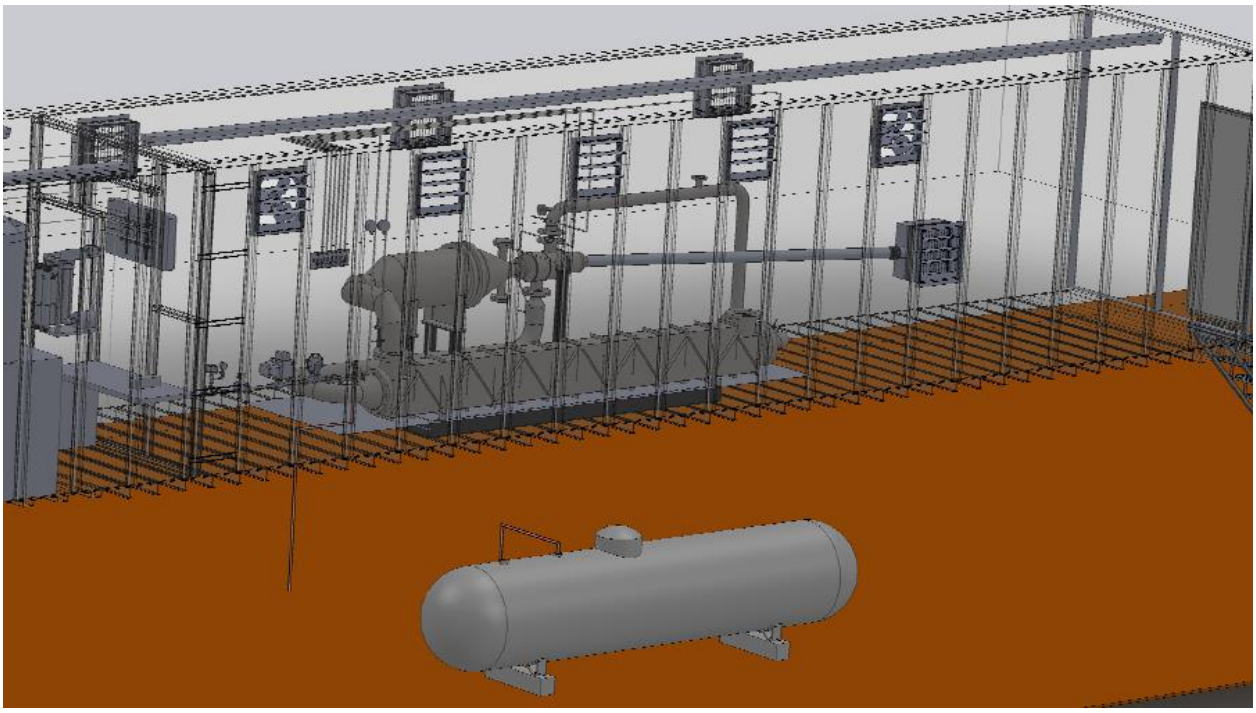


Figure 8 – CAD picture of the Hybrid Brayton system installed in a 50 ft trailer. In the picture, one can see the propane tank, the frame of the 50 ft trailer with ventilation openings and centrally located test facilities.

3.1 Power system

A Capstone C30 microturbine is selected for power generation at the HBC test facilities. This unit has a high tolerance for corrosive fluids of up to 7% H_2S and an acceptable tolerance for humidity of up to 0.05% of kg_{H2O} / kg_A [30]. The HBC test model anticipates the humidity of the working fluid to be elevated up to 0.1 kg_{H2O} / $kg_{A,}$ exceeding C30 design limits. Such application should not affect turbine performance as it will only be applied to turbine blades and the fluid is in a superheated vapour state. The Capstone C30 is capable of generating 30 kWe in a single shaft, one stage compressor/turbine/recuperator design that turns a permanent magnet generator [30]. The unit is equipped with air bearings that reduce the overall hardware size and maintenance requirements [30]. The HBC is designed to operate at pressures not exceeding 60 psig. This study did not have access to Capstone C30 detailed engineering specifications and therefore cannot predict the exact performance of the engine. The study attempts to collect experimental data to extrapolate the performance of the system and construct theoretical models

predicting efficiency behavior for a range of compression ratios (CR) and corresponding expansion ratios (ER) within the pressure limits of the system.

Figure 9 – Picture of the Hybrid Brayton system installed in a 50 ft trailer. The 4 in intake line was removed from the turbine housing for maintenance.

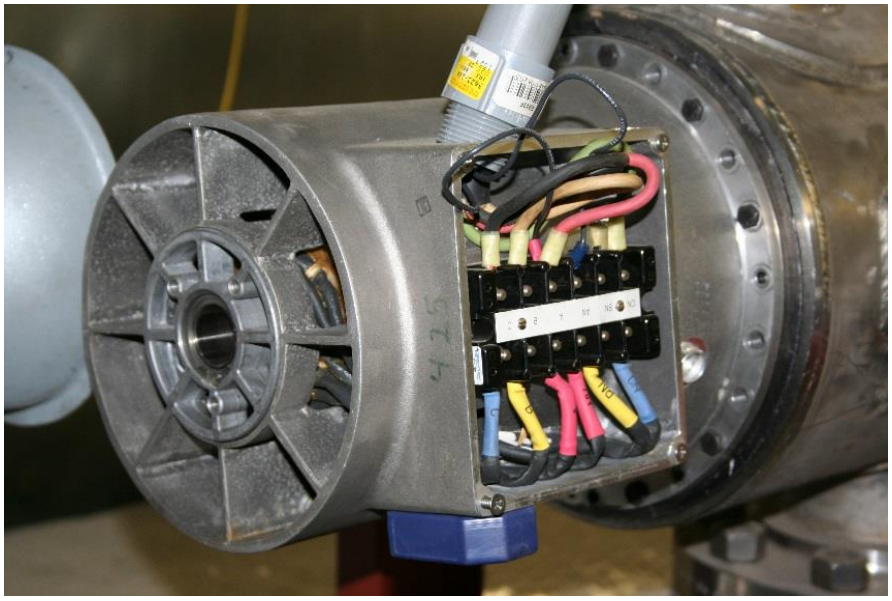


Figure 10 – Picture of C30 Capstone turbine installed in the experimental facility showing air intake, generator cover, and power connection. The turbine connects to the laboratory Hybrid Brayton cycle via the shown bolted flange.

3.2 Heat exchanger

HBC design is an externally fired Brayton cycle which utilises a heat exchanger to heat the working fluid. The main limitation of this configuration lies in an inherited thermal efficiency and a design of a heat exchanger [31]. Heat exchangers utilise the heat conduction through a media as a tool for energy exchange. This physical phenomenon is governed by the second law of thermodynamics. The design and performance of heat exchangers are well understood as they have been utilised for hundreds of years. Generally speaking, one of the factors that affect heat exchanger efficiency is the surface area of the media. Increase in media's surface area has an associated cost both in term of material economy and pressure losses due to drag affects. In addition, heat exchanger performance drops over its life cycle due to fouling, resulting in additional pressure drop, erosion and reduction of media's heat conductivity characteristics. For these reasons externally fired Brayton cycles found some experimental success in medium and large power generation applications and more limited application in a smaller scale setting [31].

Design of the heat exchanger shown in Figures 11 to 14 is generally driven by the amount of heat that is required to be exchanged between the source and the receiver fluids [32]. This process is governed by the following Fourier equation:

$$Q = UA\Delta t \tag{1}$$

where

$$A = h_w l_w \tag{2}$$

$$\Delta t = T_H - T_L \tag{3}$$

And

Q	Heat energy
U	Overall coefficient of heat transfer or inverse of heat resistance
A	Surface area over which the heat transfer is occurring
Δt	The temperature difference between the source and receiver
h_w	Height of the wall
l_w	Length of the wall

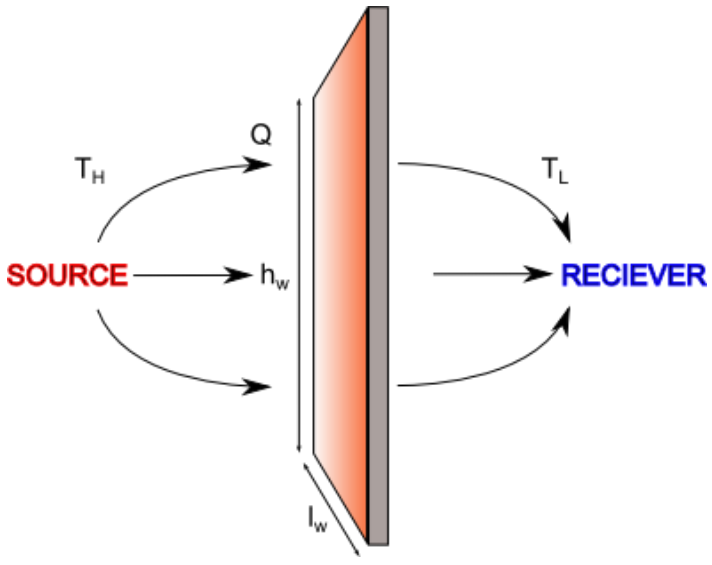


Figure 11 – Schematics of energy flow during a heat transfer from hot source to a cold receiver

The heat resistance consists of four distinct components that govern the rate of heat energy transfer:

$$R_T = \frac{1}{U_T} = \frac{1}{h_s} + \frac{1}{h_r} + r_w + r_f \quad (4)$$

where

R_T	Total heat resistance of a system
U_T	Overall heat transfer coefficient
h_s	Heat transfer rate of the source fluid
h_r	Heat transfer rate of the receiver fluid
r_w	Heat flow resistance of the wall
r_f	Heat flow resistance of fouling material on inner and outer walls

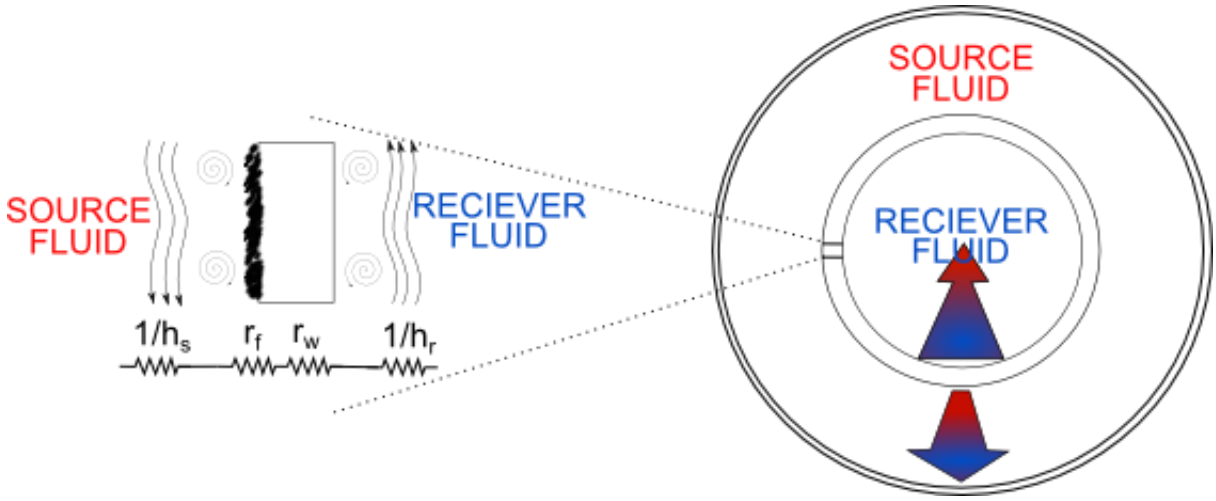


Figure 12 – Schematics of factors contribute to total heat resistance of a counter flow heat transfer [32]

Although there are four distinct variables that govern the heat transfer between fluids, there are two heat transfer processes that address these components, convection and conduction [32]. Conduction is responsible for all the heat transfer that occurs in a solid median, while convection addresses heat transfer in the gas medians [32]. The double pipe heat exchanger principle, demonstrated in Figure 13, is used in the HBC's test facility and will show a Logarithmic Mean Temperature Difference (LMDT) between the fluids in the shell and tube.

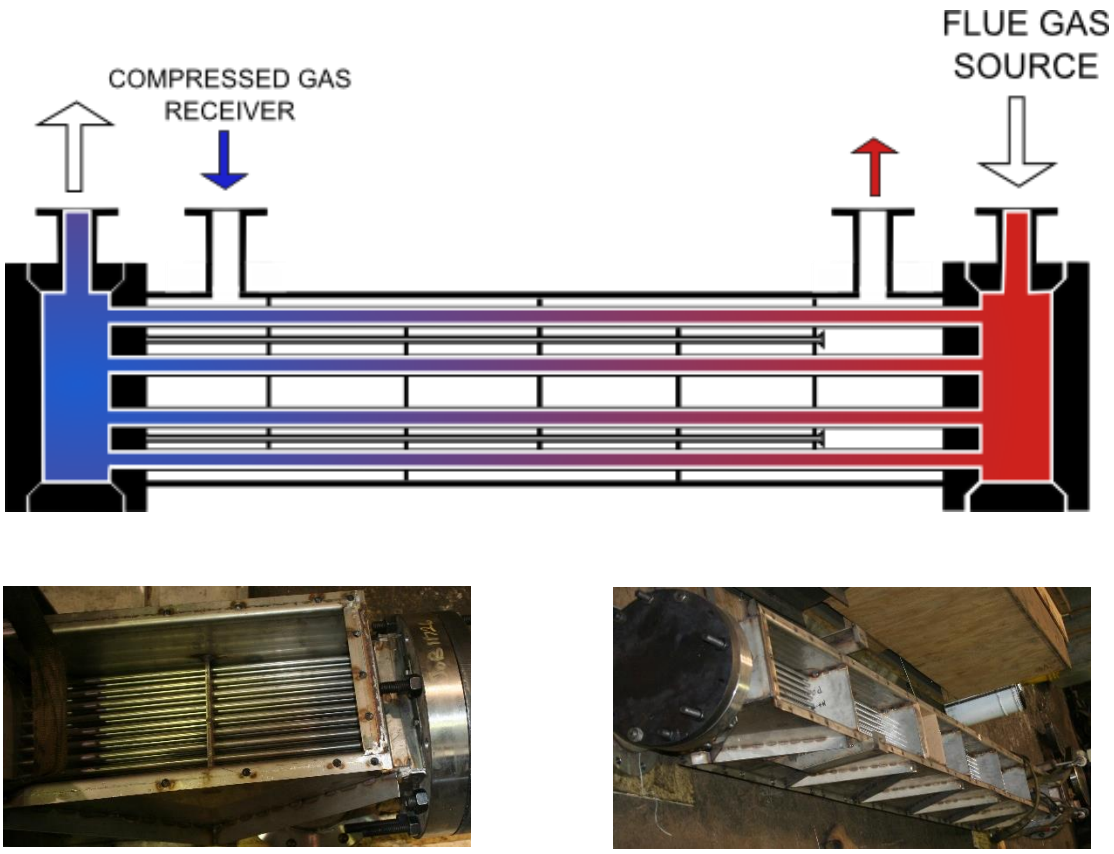


Figure 13 – Fix head tubular heat exchanger and two picture examples of the heat exchanger used for the laboratory HBC. Pictures taken during contribution.

This design configuration is chosen as an optimal compromise between simplicity, heat transfer rate and manufacturing cost. An alternative design may improve the overall efficiency of the system but at this preliminary stage of experimental design, the overall proof of the concept within the funding envelope available was the driving factor behind this design. Some of the limiting factors that are considered for this design, and which must be considered for future system modifications, are the mobility of the HBC, fabrication and material cost, fouling deposition effects and thermal expansion.

Operational heat exchangers suffer from deposition of debris on the heat transmitting surfaces over the service life of the system [33]. Such films of debris are non-metallic in nature and have a low heat transfer coefficient [33]. Due to its heat transfer characteristics, it is common in the design of HEs to overdesign the U_T of the exchanger in order to compensate for the effects of

fouling [33]. A fouling factor is introduced as part of the U_T in order to address that phenomenon for equipment sizing purposes. The R_f value is more of an educated guess because it is affected by a fluids composition, heat exchanger material, heat exchanger operational temperature, fluid turbulence throughout the heat exchanger and time in service. It is evident that the factors mentioned above are very difficult and practically impossible to determine during the design phase. Literature recommends a fouling factor of $0.001\frac{\text{hr}\cdot\text{ft}^2\cdot^\circ\text{F}}{\text{Btu}}$ for steam and exhaust and $0.002\ \frac{\text{hr}\cdot\text{ft}^2\cdot^\circ\text{F}}{\text{Btu}}$ for air [33]. In the HBC design, a fouling factor of $0.001\frac{\text{hr}\cdot\text{ft}^2\cdot^\circ\text{F}}{\text{Btu}}$ is used during the design phase. Close observation of HBC performance over time will determine the required frequency of heat exchanger maintenance and cleaning in order to minimize the effect of the inherent fouling factor. The design of the heat exchanger utilises the shell side for clean air movement and tubes for hot flue gases. This design will help with cleaning of the heat exchanger if excessive fouling occurs.

The HBC heat exchanger design accommodates regular thermal expansion/contraction of the system within the proscribed range of temperatures. This consideration limits the type of heat exchanger that can be used to withstand the thermal stresses generated during operation of the cycle.

The engineering considerations for heat exchanger design mentioned above are not all inclusive in nature and other factors may be considered in the future. The heat exchanger is a focal component of the HBC efficiency and as such, presents an opportunity in the future to balance thermal efficiency with economic factors which include mobility, maintenance and initial cost.

The governing equations that are used to design and study the pressure drop and the fluid heat transfer coefficient of the receiver and working fluids in the heat exchanger are described by the set of formulas and graphs found in D.Q. Kerns “Process Heat Transfer” book [34].

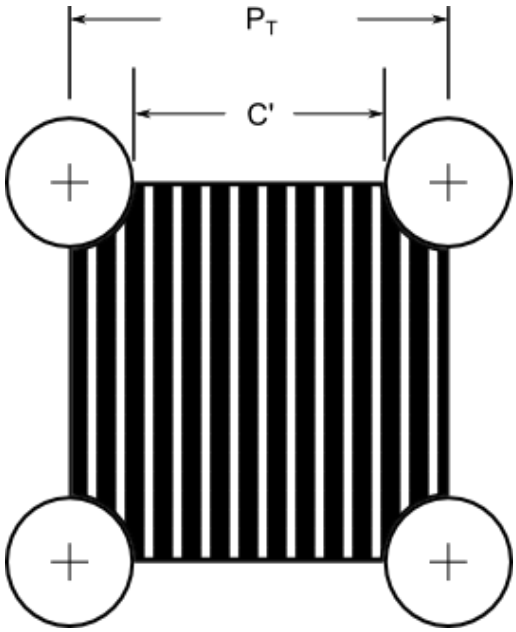


Figure 14 – Diagram of square pitch dimensions used to determine equivalent diameter [34]

$$D_e = \frac{4 \text{ x Free Area}}{\text{Wetted perimeter}} \tag{5}$$

$$d_e = \frac{4 \text{ x } (P_T - \pi d_o^2/4)}{\pi d_o} \tag{6}$$

Kern provides a set of graphs that facilitate a practical approach to heat exchanger design [34]. These graphs describe the heat transfer coefficients and friction losses within the tubes (receiver fluid) and shell (source fluid) as follow:

Determine heat transfer rate h_i within the tube:

$$j_H = \frac{h_i D}{k}\left(\frac{c\mu}{k}\right)^{-\frac{1}{3}}\left(\frac{\mu}{\mu_w}\right)^{-0.14} \tag{7}$$

$$Re = \frac{DG}{\mu} \tag{8}$$

Determine pressure drop within the tube:

$$\Delta P_t = \frac{f \times G_t^2 \times L \times n}{2 \times g \times \rho \times D \times \varphi_t} = \frac{f \times G_t^2 \times L \times n}{5.22 \times 10^{10} \times D \times s \times \varphi_t} \text{ psi} \quad (9)$$

Determine heat transfer rate h_o across shell bundle, utilising Equation 8:

$$j_H = \frac{h_o D_e}{k} \left(\frac{c\mu}{k}\right)^{-\frac{1}{3}} \left(\frac{\mu}{\mu_w}\right)^{-0.14} \quad (10)$$

Determine the pressure drop across the shell:

$$\Delta P_t = \frac{f \times G_t^2 \times D_s \times (N+1)}{2 \times g \times \rho \times D_e \times \varphi_s} \text{ psi} \quad (11)$$

All equations used in this section are empirical in nature. The equations' result is interpolated from the graphs in the Kern's book [34]. The results converted to standard units in order to maintain consistency with the other aspects of the study.

The following assumptions were made during the design and simulation of the heat exchanger:

- Prandtl number is assumed to be constant at $Pr = 0.74$ both for source and receiver fluids.
- Atmospheric air density is assumed to be constant at $\rho_{atm} = 0.073 \ lb/ft^3$.

3.3 Propane fuel system

The energy source for the HBC can be of any nature and is only limited by the external burner/combustor design. For HBC commission experiments addressed in this study, a selection is made to utilise propane gas as a fuel source in order to validate experimentally the available theoretical data.

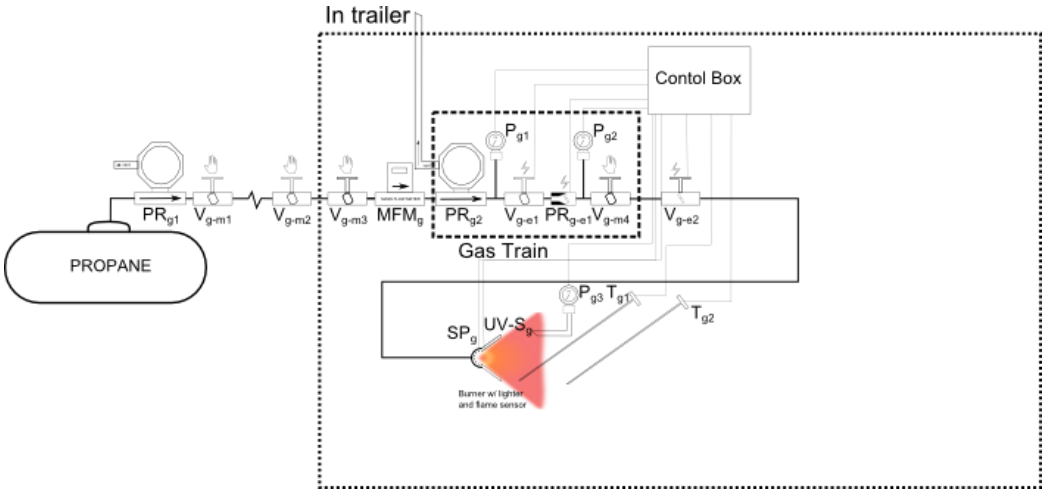


Figure 15 – Schematics of Hybrid Brayton cycle test facility propane gas supply system. Two thermocouples and a positive pressure interlock are used for control and safe operation of the setup.

All gas train components shown in Figure 15 of this system are designed by Maxon burners. The HBC test facility uses a Maxon AIRFLO type duct burner. The system installation requires a certified gas technician and meets safety regulations. The gas system includes multi layered fail safe mechanisms and automatic shutdown of the system in case of flame blowout or loss. The user has the ability to regulate the desired combustion temperature of the mixture through the control of the temperature downstream from the burner. The MFM_g 16 Series Precision Flow Meter by Alicat Scientific allows the user to determine the combustion rate of the gas during the experiment, generating a digital data log of mass flow rate for detailed energy analysis of the system.

3.4 Pressurized water injection system

The addition of pressurized water into the HBC is the key component of the HBC power generation system that distinguishes it from others. Moderate water injection into HBC system results in the improvements in thermodynamic efficiency. The Water Delivery System (WDS) shown in Figure 16 includes water filtration using reverse osmosis, storage, pressurisation, flow monitor and injection control.

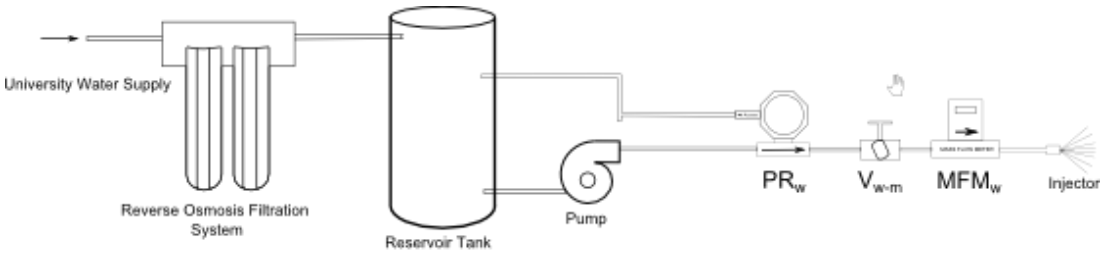


Figure 16 – Schematic of water delivery system used in Hybrid Brayton cycle test facility

Filtration and storage – The purification section of WDS in Figure 17 consist of sequential filters connecting municipal water supplies to a storage tank. The water supply from the municipal tap passes through a course filter followed by a three stage reverse osmosis filtration system. These filters are designed to eliminate any water impurities in an effort to reduce sediments in heat exchanger and turbine over prolonged HBC operation. The reverse osmosis filtration rate is higher than the maximum injection rate used by the HBC system.

Figure 17 – Picture of Hybrid Brayton cycle test facility water injection system filtration and storage section

The water storage tank shown in Figure 17 is capable to hold approximately 300 litres of water. WDS offers the ability to operate HBC for several hours with no water supply on site.

High Pressure Pump – An electric centrifugal pump, Hydro Cell P200NPTSS015C shown in Figure 18, pressurizes and pumps water from the reservoir tank to the HBC injection nozzle at pressures between 60 – 150 psig. The pumping rate is manually set by the HBC operator with a JNEV - Micro Drive by Teco frequency control unit installed in the HBC's control room, as shown in Figure 19. The electric power drawn by the pump will depend on the amount of water required to operate the system. The electric engine, Balder IDNM 3534, that operates the pump is rated for 1/3 horse power at 76% efficiency, maximum power draw on the system will not exceed 0.326 kWh which is about 1% of the potential power generation of this system.

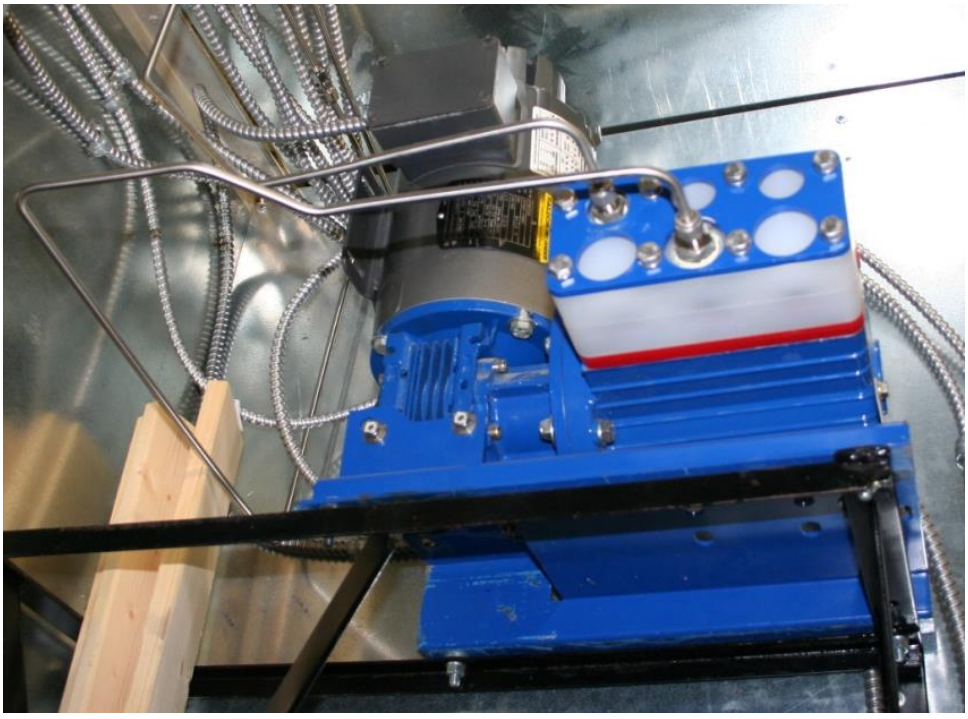


Figure 18 – Picture of Hybrid Brayton test facility water pump

Figure 19 – Picture of Hybrid Brayton test facility water pump manual control console

The Vm-w is the manual valve to regulate water flow to the injection head. The pressure regulator PRw between the pump and the valve releases the excess flow back to the reservoir tank.

Figure 20 – Picture of water valve and water flow meter in circles with the gas flow controller showing in the middle

Flow meter - The MFM_w is a 16 Series Precision water flow meter by Alicat Scientific (Figure 20). The mass flow rate information feeds to the Compact Rio data acquisition system.

Injector – Two BETE nozzles available for this study, P32 and P20, as shown in Figure 21. The number following the letter P indicates the corresponding orifice size of the nozzle, i.e. P20 has orifice diameter of 0.020 inch. With experimental results the optimal head selection will be made to conduct target experiments. This research does not address the effects of nozzle type on the thermodynamic properties of the system.

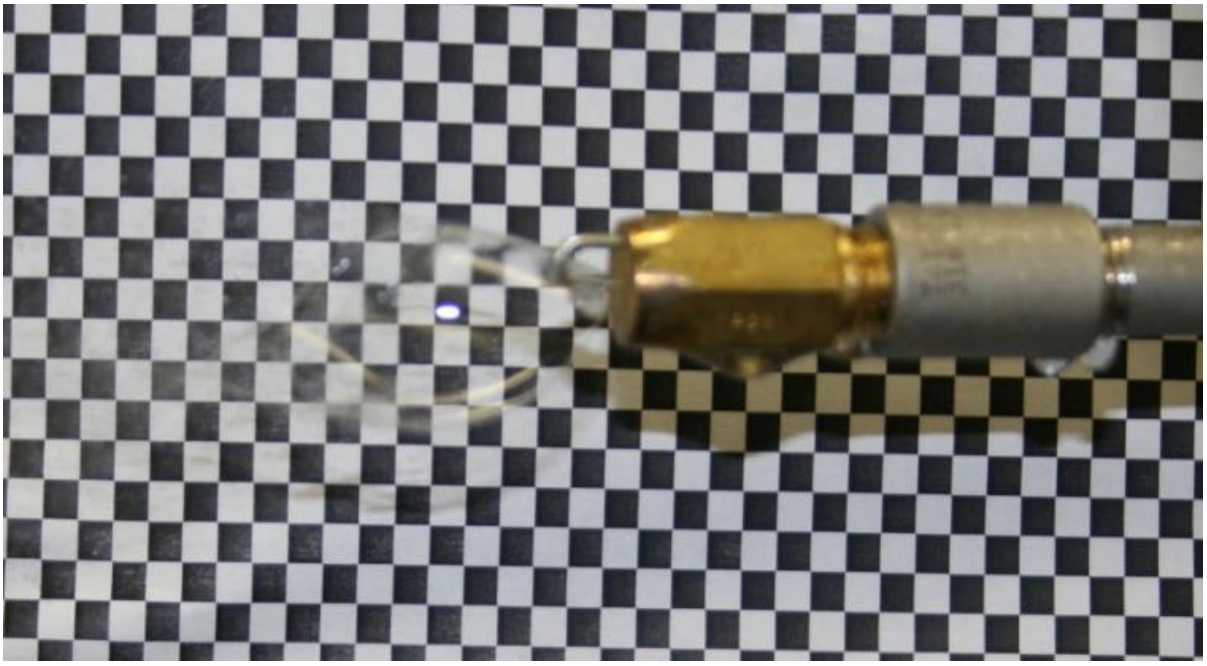


Figure 21 – Picture of BETE water injection nozzle in operation in ambient air with mist showing

The science of mist generation identified a range of measuring tools in order to define the droplet size and nature of the mist. Arithmetic mean diameter, surface mean diameter, volume mean diameter and sauter mean diameter are some of the most common definitions used in mist generation [35, 36]. Figures 22 and 23 outline the performance of the BETE nuzzles at set pressures.

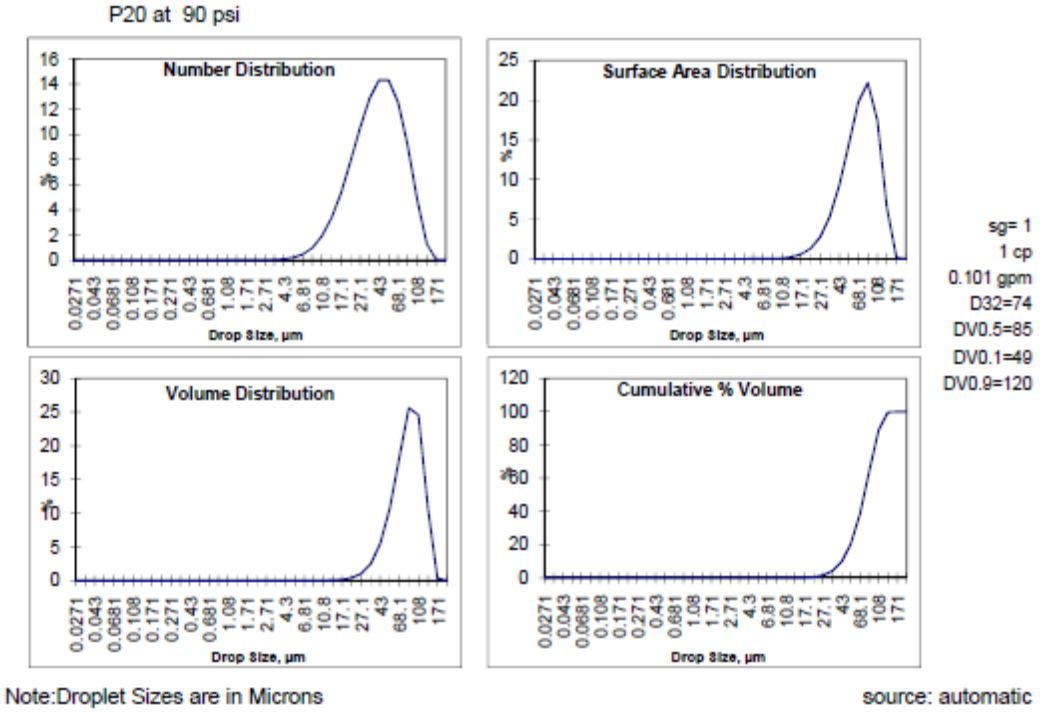


Figure 22 – BETE P20 @ 90 psi injection stream profile [36]

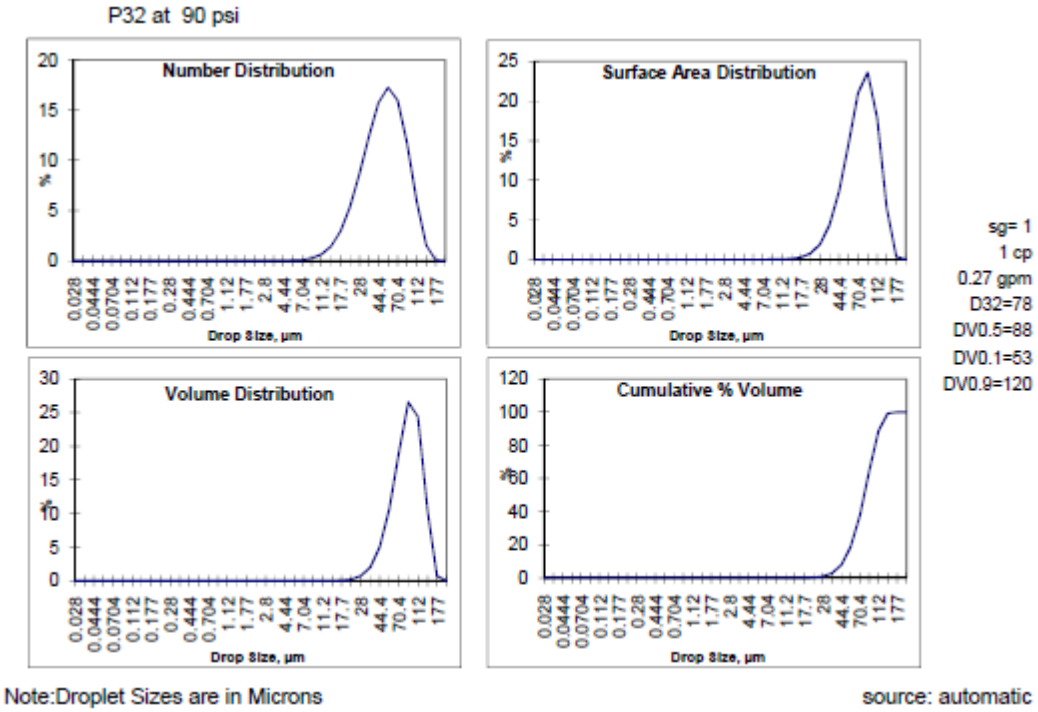


Figure 23 - BETE P32 @ 90 psi injection stream profile [36]

It is evident from these figures that the size of the droplet varies significantly throughout the water cone. Introduction of the jet cone into a hot medium is difficult to model, as the temperature profile of the medium, kinetic and moment forces and its interaction with surrounding droplets will all contribute to the evaporation rate [35]. To further complicate the HBC water injection system, it occurs in an enclosed four inch pipe, which is responsible for creating additional turbulence and mixing affects.

Both the simulation and the experimental facility setups focus on determining the overall efficiency of using evaporative cooling. This is done by determining the mass flow rate and thermodynamic state of primary air intake and water injection before and after mixture. Any thermodynamic interaction on a particle level is outside the scope of this study. Any deviation between the models and the test results could be explained by incomplete evaporation resulting from poor mixing or insufficient thermal availability to phase change the water. During future experiments this can be confirmed by checking the bottom entry flange to the heat exchanger, where water may pool if this situation exists. For the purposes of this study, the assumption is made that the ideal mixing with complete evaporation occurs.

The governing parameter that limits the rate of water injection is the efficiency of the heat exchanger and the saturation curve during the fluid expansion in the turbine. The addition of water into the working fluid of the HBC system increases the efficiency potential of the cycle, allowing higher energy transfer rate at the heat exchanger. This occurs due to the increase of mass flow through the heat exchanger and an increase in the differential temperature between hot and cold flow in the heat exchanger. One of the limiting factors is the thermodynamic capacity of the working fluid to completely saturate the water after injection. Excessive water injection can lead to the expansion of the moist air in the turbine where drop in pressure and temperature occurs due to the conversion of thermal energy to mechanical. An excessive concentration of water particles in the working fluid (prevalent during system start up) will reduce the TIT and consequently may trigger a water condensation during the expansion. This phenomenon may result in wear to the turbine blades and therefore is undesirable. However, this condition is avoidable by the system controller.

3.5 Electrical

The HBC test facility produces a 3 phase AC electric current of up to 30 kW at 500 to 600 V and approximately 1650 Hz [30], as shown in Figure 24. The power is converted to 3-phase DC using Diode Bridge. This experiment anticipates generating 12 kW_e due to the system being built without a recuperator and having only one injector. This experiment is designed to measure the electric output of the HBC as it discharges the power into a heater load bank. The load is induced by one 5 kW and one 10 kW industrial electric heaters that were modified to take in high frequency load. The heaters fan remains on the power grid in order to ensure that cooling of the heaters is maintained and those units can perform over an extended period of time in order to avoid premature failure. Each load is engaged remotely from the control room and all are connected in parallel to the 3 phase output of the C30 capstone turbine.

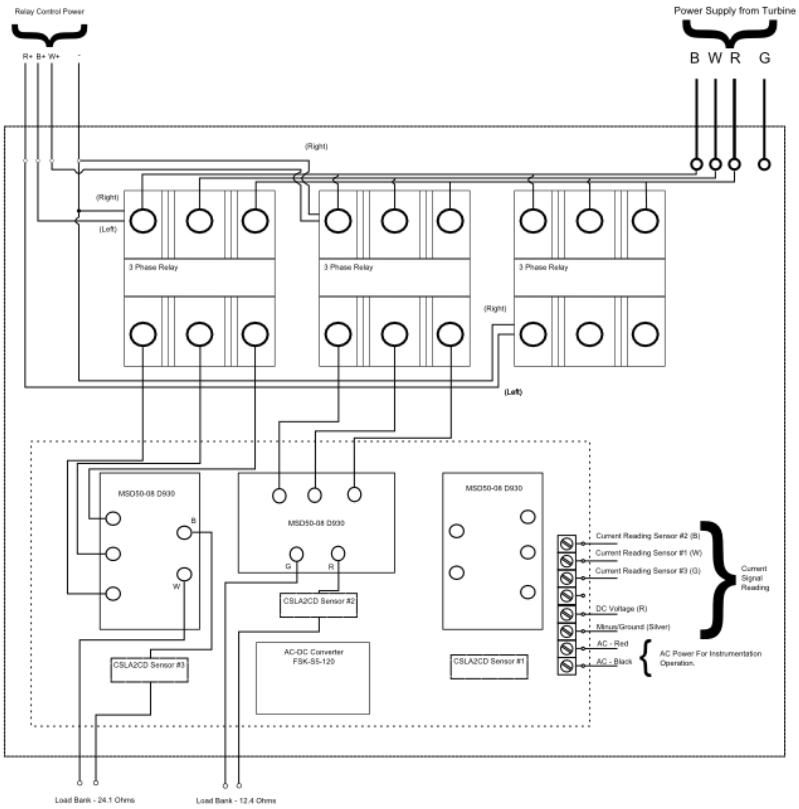


Figure 24 – Schematics of the Hybrid Brayton test facility's actual electric load

This experiment did not address the rectification of the electric power to 60 Hz, 600 V, 3-Phase electricity due to budgetary constraints. Such systems are commercially available and can be adopted at later stages of commission or at the production phase.

3.6 DAQ system

The Texas Instrument's Compact RIO NI CRio - 9014 is used to measure and condition the signals from the HBC test facility sensors. The LabView customized program shown in Figure 25 processes the data and outputs information containing; seven pressure samples, thirteen temperature samples, one humidity sample, two mass flow measurements, and one electric frequency, voltage and current sample.

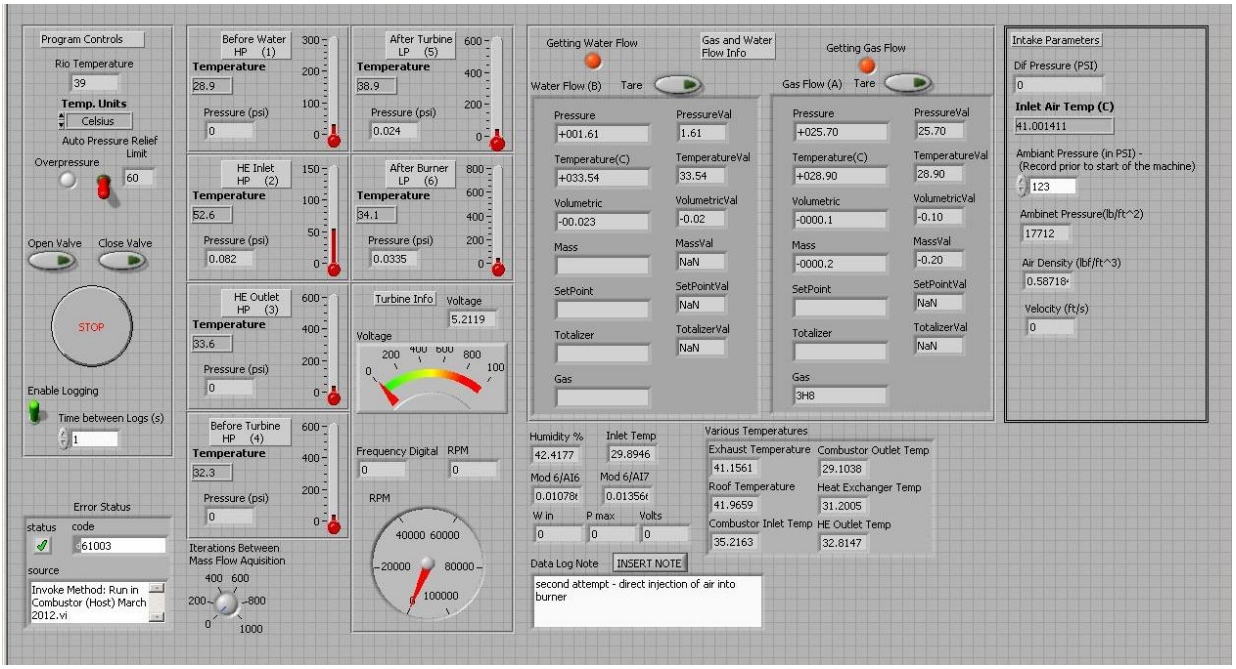


Figure 25 – Picture of customized Hybrid Brayton test facility control panel interface for system control and data recording

3.7 Operational limits

Test facility shell – The HBC system is custom constructed from 304H stainless steel and certified for operation with pressures of up to 60 psig and temperatures of 1350°F. Pressure and

temperature sensors are designed and installed in predetermined locations to ensure optimal thermodynamic reading at every stage of the HBC.

Pressure Sensors – There are three types of Honeywell Sensotec pressure transducers found in the HBC facility: (1) six FPG model with range 75 psig, (2) one psig model with power supply 9-28 V_{DC} supply and 4-20 mA output, and (3) a PFD model differential pressure 1 psi range with 9-28 V_{DC} supply and 4-20 mA output. All these sensors connect to the CRio module NI 9203.

Thermocouple Sensors – There are two types of K-type thermocouples configurations used in the test facility. One set takes a reading of the working fluid and the structure surface temperatures and connects to the NI 9211 module. In addition, there is one thermocouple in the compressor inlet measuring temperature and humidity which is part of the Rotronic HydroFlex 4 humidity temperature transmitter, which connects to the NI 9205 module.

Mass Flow meters – The 16 series precision flow meters by Alicat Scientific are both connected directly to NI CRio-9014 via eight pin serial connectors.

Frequency and Voltage – Sampled directly from one of the phases of electric outlet with a 100/1 voltage divider and comparator as demonstrated in Figure 27.

The facility was insulated after the cold flow commissioning tests performed. The insulation is designed to allow people to be in the lab but remains somewhat warm if the insulation is

touched.

Figure 26 – Picture of the HBC insulation installation

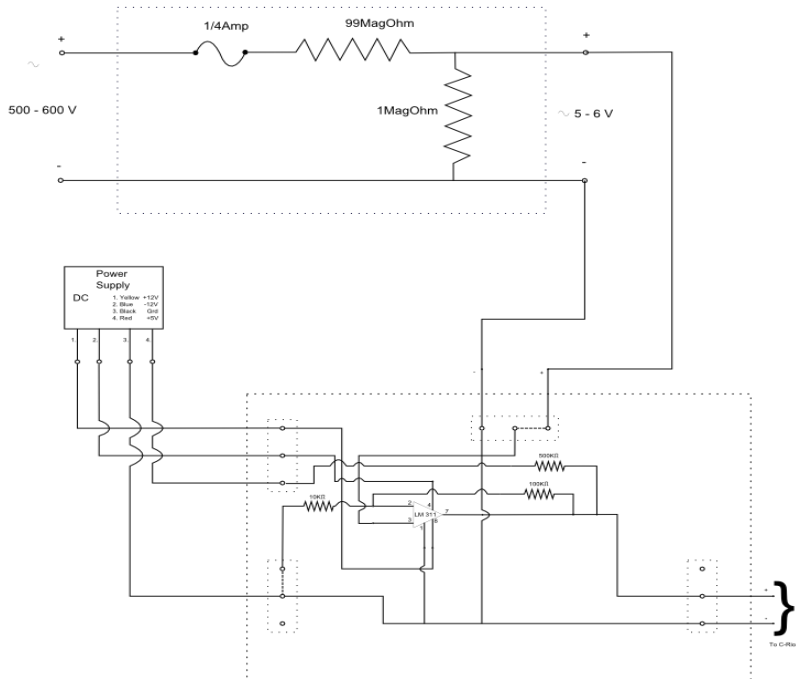


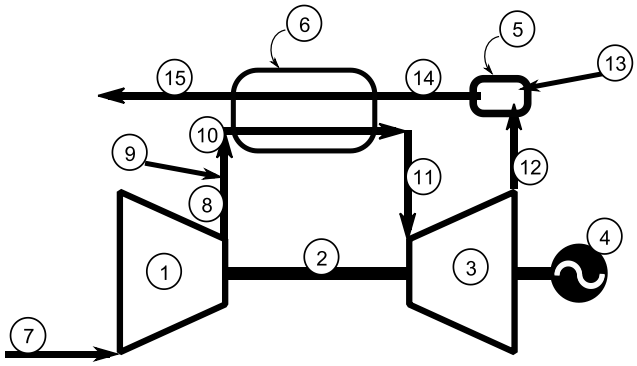
Figure 27 – Schematics of Hybrid Brayton cycle test facility voltage sampling device

The outlet is then connected to NI 9205 and NI 9401 for frequency and voltage sampling.

Current indicator – the HBC is set up with three inductive analog current sensors, Honeywell CSLA2CD, connected to NI 9205.

The location and accuracy of the instruments provided by the manufacturer are shown in Table 2. High accuracy instruments are also provided with calibration information that is used to adjust the output of the sensor.

Table 2 – Summary of instrumentation found in Hybrid Brayton cycle test facility



Loc	Sensor Type	Model	Input Range	Output range
7	• Humidity/Thermocouple • Differential pressure	• HydroFlex 4 • PFD	• 0 to 50°C •	• 0 – 20 mA • 4 – 20 mA
8	• Thermocouple • Pressure Sensor	• K type • FPG	• -200 to 1250°C • 0-75 psig	• -80mV to 80 mV • 4-20 mA
9	• Mass flow /Temp/Pressure	• 16 Series	• P 0-145 psig • T -10 to 50°C • M 50 SLPM	• 0-5.0VDC • 0-5.0VDC • 0-5.0VDC
10	• Thermocouple • Pressure Sensor	• K type • FPG	• -200 to 1250°C • 0-75 psig	• -80mV to 80 mV • 4-20 mA
11	• Thermocouple • Pressure Sensor • Surface Thermocouple	• K type • FPG • K type	• -200 to 1250°C • 0-75 psig • -200 to 1250°C	• -80mV to 80 mV • 4-20 mA • -80mV to 80 mV
12	• Thermocouple • Pressure Sensor • Surface Thermocouple	• K type • FPG • K type	• -200 to 1250°C • 0-1 psig • -200 to 1250°C	• -80mV to 80 mV • 4-20 mA • -80mV to 80 mV
13	• Mass flow /Temp/Pressure	• 16 Series	• P 0-145 psig • T -10 to 50°C • M 250 SLPM	• 0-5.0VDC • 0-5.0VDC • 0-5.0VDC
14	• Thermocouple • Pressure Sensor • Surface Thermocouple	• K type • FPG • K type	• -200 to 1250°C • 0-75 psig • -200 to 1250°C	• -80mV to 80 mV • 4-20 mA • -80mV to 80 mV
15	• Thermocouple • Pressure Sensor • Roof Thermocouple	• K type • FPG • K type	• -200 to 1250°C • 0-75 psig • -200 to 1250°C	• -80mV to 80 mV • 4-20 mA • -80mV to 80 mV
4	• Voltage/Frequency • Current	• NI9205/9401 • CSLA2CD	• 0-20V/0-5V • -72 to 72A	• N/A • 0 –V@ constant current

4. Results and comparisons

4.1 Test facility results

The test facility was first commissioned using cold flow and then hot flow after all piping insulated. All instruments and controls where verified during these tests.

The major roadblock during commissioning was with the mechanical start of the turbine using an external compressor. This study had limited success in obtaining experimental results from the HBC facility due to difficulty encountered in starting the turbine. Difficulties arose primarily because of the current approach and design limitations of the laboratory system. This section will explain the experiments that were conducted in an attempt to commission the facility and then provide recommendations and deductions. In order for future applications of the HBC system to be successful, it is important that the limitations of the system and lessons learned in this study are understood.

The C30 Capstone gas turbine in the HBC, like all Capstone models, is designed to start up electrically within 1 second using the permanent magnet generator operating in reverse. Grid or battery electricity is applied into a specifically designed converter that generates high frequency output onto the solenoid, forcing the turbine and compressor to spin [30]. This provides combustion air and the natural gas is fired.

For the HBC implementation, once the turbine and compressor are turning at the same speed, the resulting HBC air movement is used to light up the burner, which leads to a self–sustainable cycle and subsequently to a power generating cycle within approximately 30 seconds [30], as here the time constant of the heat exchanger delays start-up. The electric start is not available for this study as its development is too costly and an alternative approach is developed for a proof of concept. Direct compressed air injection into the HBC system was assumed to be a feasible alternative to eliminate the electric start and conceptually there was no evidence to prove otherwise. The HBC study selected a refurbished Sullivan Palatek 675H high performance air

compressor that produces 675 CFM @ 100 psi, approximately 40% above the required air flow capacity for C30 turbine.

The turbine was successfully tested in cold start mode with a similar compressor that was rented at the initial assembly of the facility during cold flow commissioning. However, later during commissioning the excess air flow capacity of the chosen external compressor produced insufficient pressure drop across the C30 turbine, causing it to stay immobile. On occasion, during testing some movement of the C30 shaft is observed with a pressure drop of approximately 17 – 18 psi across the turbine. The compressor is unable to maintain such a pressure drop across the turbine blades, and after a few seconds dropped down to 10 – 12 psi. After investigation, it is found that the external compressor output is not adequate for start–up of the HBC. It is concluded from several observations during the commissioning attempts that pressure drops in excess of 20 psi across the gas turbine will engage the air bearing and bring the shaft into motion.

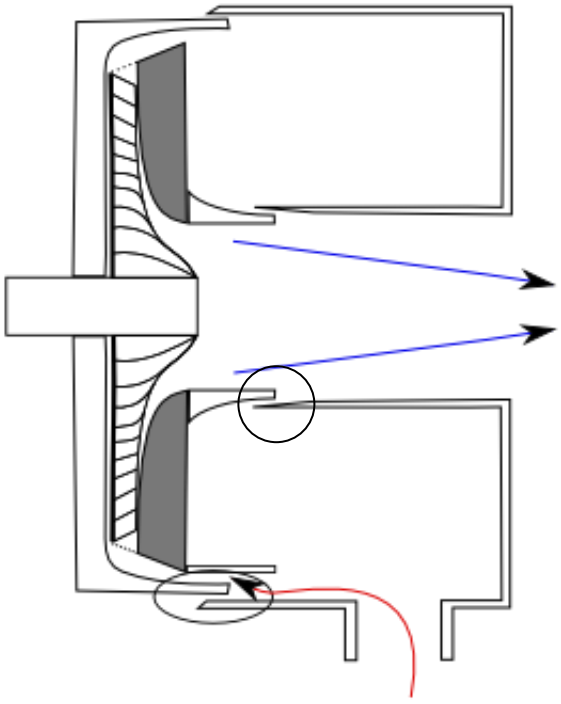


Figure 28 – Schematics of C30 Capstone turbine seated in the housing with corresponding gaps for thermal expansion. The red arrow line represents the high temperature and pressure working fluid and the blue arrow line symbolises exhausted working fluid at lower temperature and pressure relative to the inlet. These gaps are the main reason for the unsuccessful start–up of the turbine using an external compressor.

Figure 28 visually shows the reasons for pressure losses during the start–up of the HBC test rig via the external compressor. The system is designed to seal the gaps when it achieves the operational temperature of approximately 900˚F. It is important to note that the HBC operational temperature is lower than the C30 operational temperature, because the original C30 design is based on the premise of direct fuel injection. The housing design for the C30 in HBC is adopted from the Capstone conventional housing. Therefore, it is difficult to conclude at this stage of the study if 900˚F is a sufficiently high operational temperature to achieve a completely sealed housing. This issue must be examined closer once the system is operational, as this may produce losses that are difficult to detect during operation.

An experiment to preheat the externally injected flow prior it reaching the turbine was conducted. The test had to be prematurely terminated in order to prevent heat damage that might occur to the electrical components of the turbine.

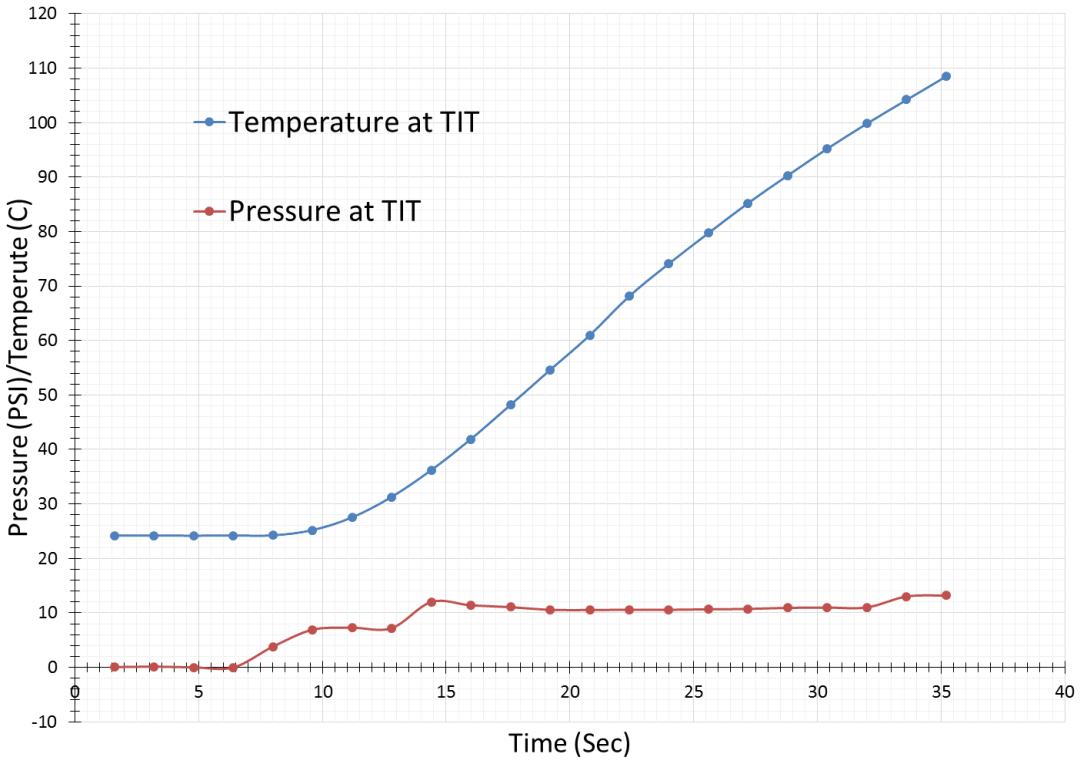


Figure 29 – Graph represent pressure and temperature reading at turbine inlet while attempting to heat up the working fluid across the turbine in order to close the gap and actuate the turbine

Figure 29 demonstrates the gage readings collected during pre-heating experiment at the turbine inlet. After lighting up the burner, the temperature of the injected air started climbing up at a rapid and steady rate. There is evidence that the expansion due to heating is reducing the gap, and that the pressure drop across the turbine starts to increase. It is anticipated that with additional time the turbine would start to spin.

The reason for prematurely stopping the experiment is the observation that the temperature of the leaking air is rising while it is travelling in the reverse direction through the compressor intake. This situation can correct itself once the shaft starts turning and the C30 compressor would start taking in air in the correct direction. However, until that state is reached, the electrical component of the generator, the coils, are exposed to temperatures in excess of 200˚C for 20 to 30 seconds at a minimum. Such temperatures would most likely melt the insulation and create short circuits as soon as the turbine would start spinning. Anticipating such a series of events, this experiment was terminated prematurely before any damage was done.

The findings from these experiments are important for future commissioning of the HBC system. There are several conclusions and considerations that must be kept in mind, prior to commencing the next phase of HBC study:

- Doubling or tripling the airflow injection in order to produce the 20 psi pressure drop across the compressor must be done with caution. The current design of HBC limits the intake of outside air to be taken in by just a two inch pipe, the flow then expands into a four inch pipe. That expansion in diameter may create undesirable resonance/vibration. Such vibration was observed at ~5% open valve in the current system, however the source of the vibration was not examined in-depth in this study.

- The screw driven compressor used for this test will inherently introduce oil into the compressed air. There is an air/oil separator system that is integral to the compressor system, but it is limited to six parts per million of air. With high flow rate compressors, this results in a large amount of oil introduced into a system. A

secondary air/oil separator was considered, but it was difficult to find high flow air/oil separators within the budget of this project.

- The core design of the C30 Capstone prescribes an electric start up for the turbine. Effort to circumvent such a start up by utilising a direct air flow injection has yet to yield results. A detailed assessment must be done in order to ensure that external air injection start-up is the correct start-up method for this test. It is important to note that the current HBC design can adopt the electric start up with minimal modification. The available secondary air supply at the test facility can still be used for the shutdown and cooling of the system.

Future studies to commission the HBC should consider the above mentioned technical problems of this system in order to learn from the experiences of this study.

Appendix B shows various procedures developed during the commissioning of the facility.

4.2 HBC simulation numerical model

There are a number of simulation models that this study assessed in order to determine the suitability for modeling the HBC system. In addition to the evaluation of these models, this study developed a custom application for the HBC simulation.

The HBC concept was developed using calculation in a Microsoft Excel spreadsheet (Figure 30). The model used the study of Nelson and Sauer [27] to provide HBC performance. The available simulation had some distinct advantages but it also presented some problems that lead this study to consider alternatives. The benefits of the Excel model are as follow:

- The model is aesthetically pleasing and allows a first time reader to understand the HBC setup, system's theory of operation and key components.

- It utilises readily available software, Microsoft Excel, which makes this program easy to share and it utilizes a familiar environment.

Some of the shortfalls with the Excel simulation are as follow:

- Iterative process is not stable and when making minor changes to inputs it often results in a system crash or an inability for it to converge.

- Model is not modular and upgrades or changes to the system must be done by one person, who knows, understands and has mastered the excel program. It is practically impossible to follow through the logical progression of the program, without any guidance from the author.

- Cannot control the calculation order that the code is executed in Excel during iterations.

- System integration with operational control of the HBC is difficult.

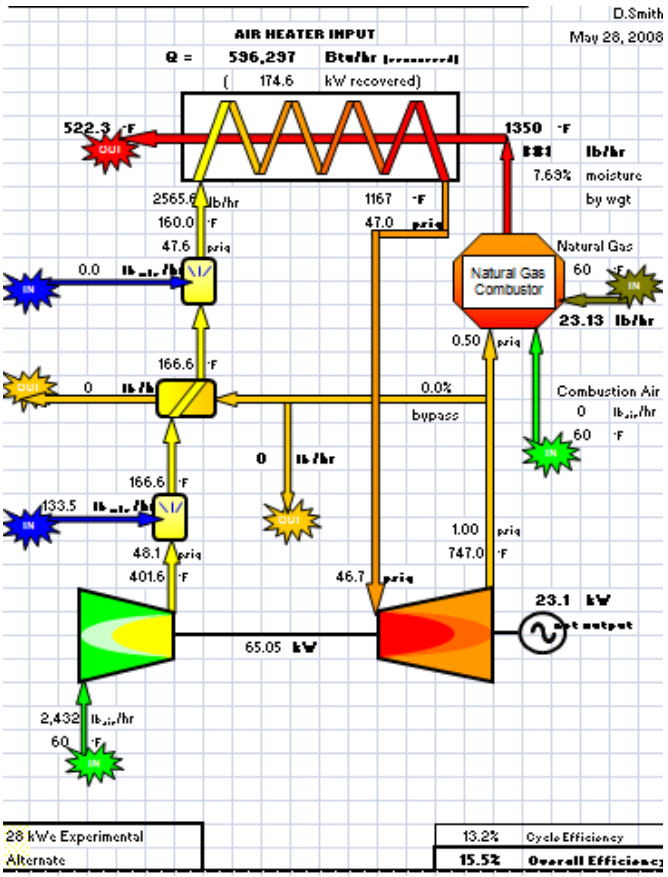


Figure 30 – Picture of initial Hybrid Brayton cycle simulation model results done in Microsoft Excel [24]

All these concerns needed to be addressed in a platform that could still maintain the interactive approach provided by Microsoft Excel. Therefore, the study selected a MatLAB platform to run the simulation. The MatLAB simulation allowed a modular simulation which provides the ability for future optimisation of HBC components independently. It also resulted in better stability of the system overall throughout the iteration process.

EUtech Thermolib add-on with simulations shown in Figures 31 to 34 provides the user with a simulation tool to simulate fundamental thermodynamic devices, chemical mixing and heat transfer systems with a wide range of customisable built in suits. Thermolib is simple to use, easy to learn and quick to compute results, regardless of the complexity of the system. The shortfall of this simulation tool is the scientific depth that was utilised to simulate different mechanical and chemical processes. The Peng-Robinson equation of state is used to determine thermodynamic properties of the range of gas substances and the IAPWS – IF97 for steam and

water is offered in the Thermolib program [37]. To understand the discrepancy in the mixing simulation between Thermolib and the other two models used in this study, a simple mixing simulation is used to demonstrate the principle difference. It mixes 1 kg/sec of air at 500 K with water at 300 K, mass flow range from 0 – 0.01 kg/sec, all at atmospheric pressure.

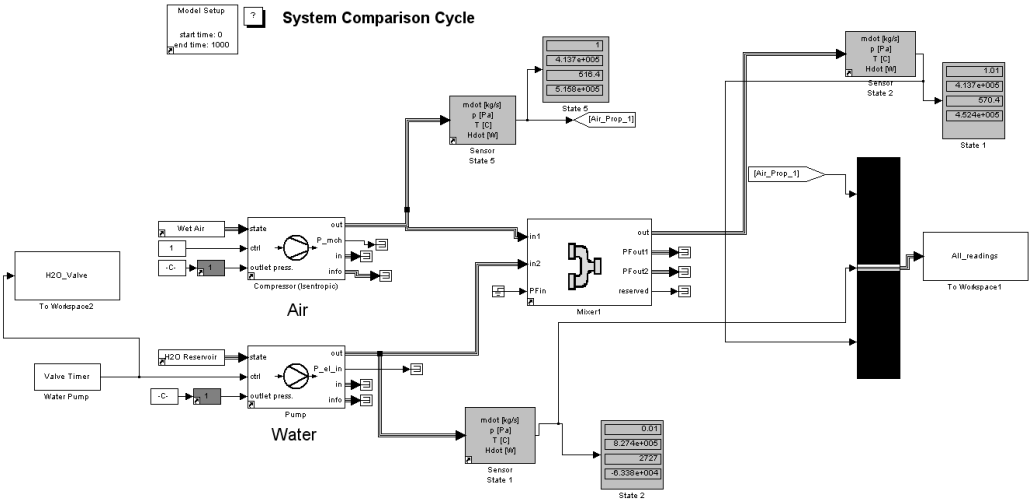


Figure 31 – Schematics of Thermolib mixture model of air and water

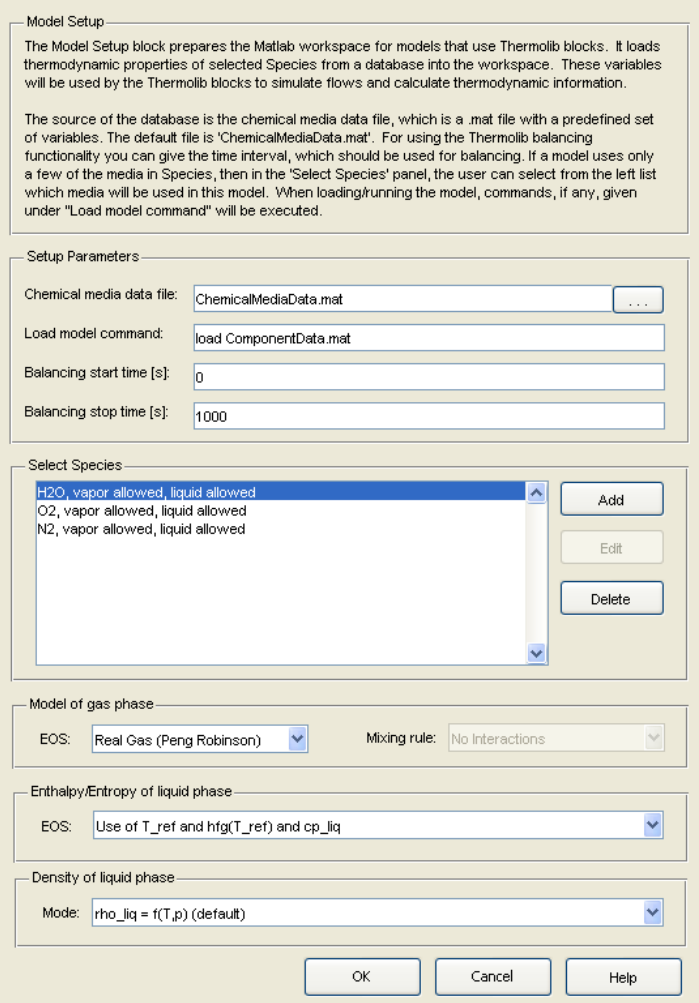


Figure 32 – Picture of Thermolib model setup window, the mixing rule option is greyed out indicating that this capability is not yet possible

The model also assumes and simulates all its fluids in the form of ideal fluids and in equilibrium state. The mixer device used in the simulation is the most appropriate for air/water mixing. It utilises ideal mixing principles [37], however it fails to deliver correct results for the mixture of air/water. The conclusion is evident from Figure 33 as we can see a linear behavior that is consistent with ideal gas mixing. This simulation tool is limited to simulating pure water or ideal gas mixing and cannot incorporate two different computational methods in a single simulation.

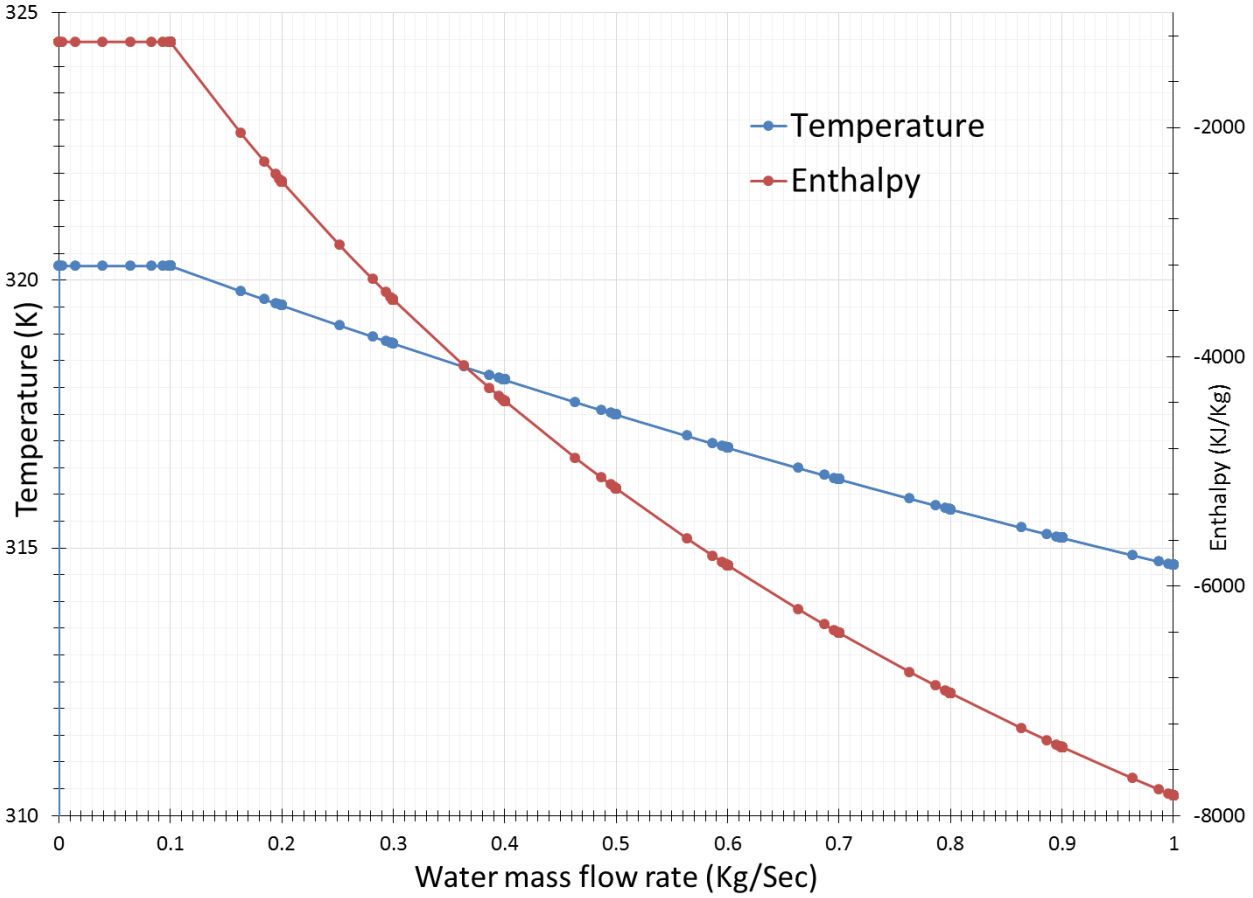
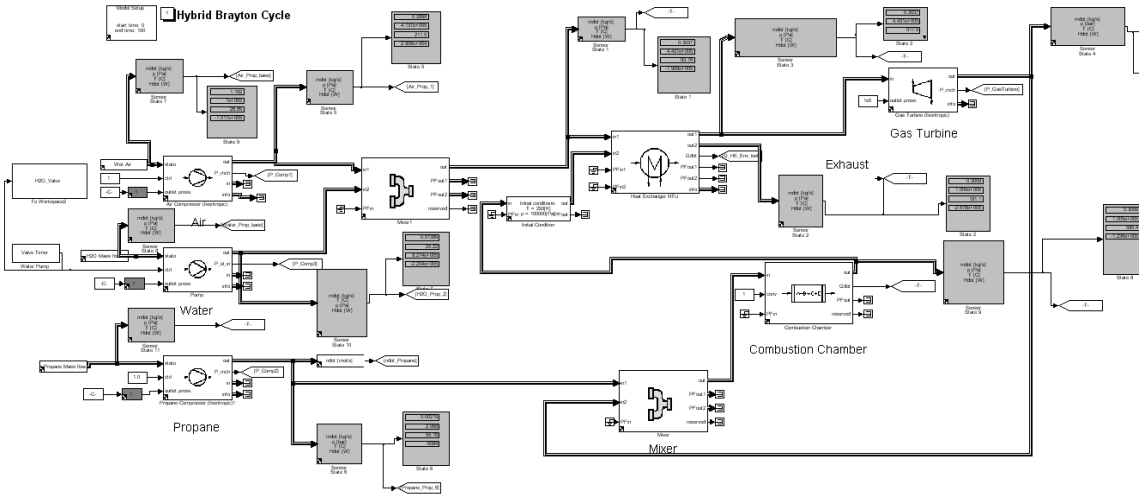


Figure 33 – Thermolib results of air/water mixture. Mixing 1 kg/sec of air with 0 – 1 kg/sec of water. No saturation region is observed on the graph.

Thermolib simulation v5.1.1 is not built to generate thermodynamic results for saturated region of liquid and gas. It is observed from Figure 32 that the Thermolib developers are considering incorporating the mixing affects in future upgrades, but with the current version this model cannot simulate the subject study.



s

Figure 34 – HBC MatLAB simulation model in Thermolib: EUTech Scientific Engineering, Thermodynamic Systems Library [CD-ROM],v 5.1.1 EUTech, Aachen, Germany [2011]

In an effort to address the existing problems with the HBC excel simulation and construct a platform in which further model development can be done, the HBC simulation [24] is coded in MatLAB. The goal is to develop a model that allows each of the HBC components within a simulation to be modular with the intent of enabling individual optimisation of each component. The approach has better potential to be used to improve the performance of the HBC and converge. The HBC simulation utilises an iterative method incorporating the Nelson and Sauer [27] equations with their original and improved coefficients. The code can be found in Appendix C. To validate the model with limited experimental data in the temperature range required, an alternative humid air properties model proposed by Herrmann et al. [26] is also compiled in the MatLAB environment. Thermolib is not used in the MathLAB model.

Before attempting to simulate a HBC model, an air/water mixture model is developed for two MatLAB simulations. The Nelson and Sauer simulation model, denoted by "O" for Original, and Herrmann et al. publication, denoted "G" for German, in the Figure 35. To understand the difference in the behaviour of the two models, a simple mixing simulation is developed to demonstrate the principle results. Both models mix 1 kg/sec of air at 500 K with water at 300 K, with the mass flow range being increased from 0 to 0.6 kg/sec at atmospheric pressure.

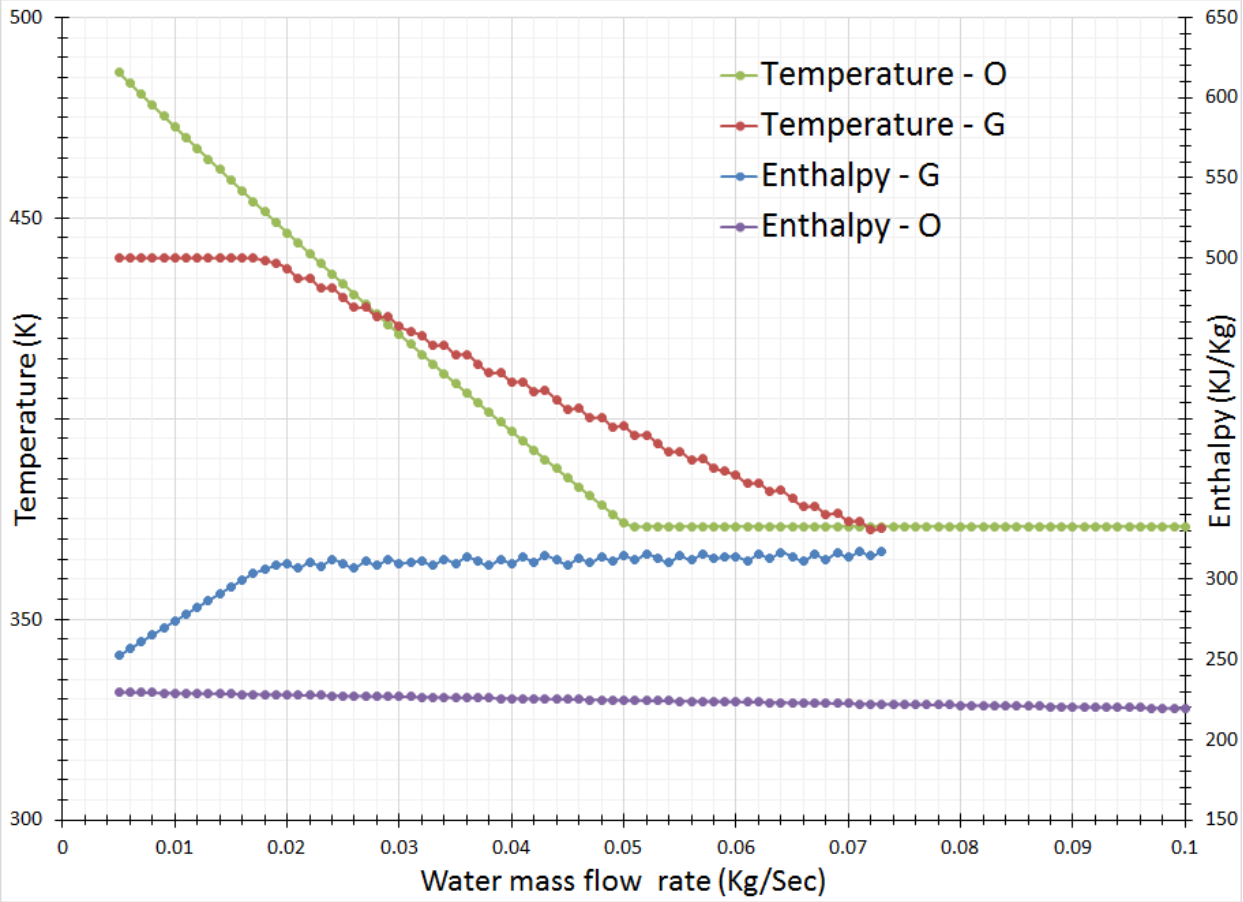


Figure 35 – Thermodynamic behaviour of air/water mixture simulated by Nelson and Sauer model (denoted "O") and Herrmann et al.model (denoted "G"). G – model stops at the saturation due to model limitations

There are some observations that should be kept in mind prior to looking at the behavior of the HBC. The base enthalpy (h°) and entropy (s°) that is used in two different studies is of a different value. For that reason this study will review *only* the trends of these simulations. We see from Figure 34 that the temperature of both temperature lines stagnate once the mixture reaches the saturation state, as predicted. The G model reaches the saturation temperature with water mass flow at 0.073 kg/sec and the O model at 0.051 kg/sec, overall that is just over 2% difference between the two, with relation to the mass flow of air. This suggests that both models, despite different fundamental approaches, should generate sufficiently similar behaviour.

The difference in enthalpies of the two models can be explained by their fundamentally different approach. The O model utilised the virial coefficient of state equation:

$$Z = \frac{P\bar{v}}{\bar{R}T} = 1 + \frac{B_m}{\bar{v}} + \frac{C_m}{\bar{v}^2} + \cdots \quad (5)$$

The virial coefficients amend the ideal gas equation of state to bring it closer to the realistic behavior of the substance at any given temperature and pressure [27]. These virial coefficients are temperature dependant and can be easily computed numerically.

The G model utilises Helmholtz energy equation to determine the thermodynamic properties of the mixture fluid [26]:

$$\frac{a_a(\rho_a, T)}{R^{Lem}T} = \alpha_a(\delta_a, \tau_a) = \alpha_a^o(\delta_a, \tau_a) + \alpha_a^r(\delta_a, \tau_a \quad (5)$$

$\alpha_a^o(\delta_a, \tau_a)$ - Ideal Gas contribution
$\alpha_a^r(\delta_a, \tau_a)$ - Residual Gas contribution

It's worth noting that the controlling variables in the Helmholtz equations are density and temperature. However, the HBC model operates using pressure and temperature variables. In order to perform HBC simulations of the new power cycle, this study incorporated IAPWS -97 solver to determine the corresponding density to the given pressure/temperature values. This correction helps to stabilise the G – model simulation.

There are some assumptions that are made prior to the integration of the G and O models in the MatLAB environment. These assumptions are designed to bring the models closer to the realistic behaviour of the system.

Pressure effects and losses – The initial conditions for modeling these studies is set at sea level pressure of 101.325 kPa with 50% humidity. Acceptable initial values for pressure and humidity in these models are those above the crystallisation of water.

The default Compression Ratio (CR) of the turbine is assumed to be 4.49. This assumption is made in the initial simulation model [24] and is consistent with a high end performance range of single stage compressors [30] [38]. There are however studies that suggest that this assumption is generous and the C30 Capstone only delivers a CR of 3.5 [38]. The HBC is not designed to operate exclusively with the C30 Capstone. For that reason, the simulation reviews the performance of different compression ratio ranges.

The pressure drop is assumed to be ½ psi between stages 10 and 11 of the HBC, shown in Figure 7. This assumed pressure loss addresses the notable effects of turbulence and drag losses that occur in the system. The pressure at stage 12 is presumed to be ½ psi above ambient pressure. This is derived from the fact that the burner requires a minimum ½ psi pressure drop across it, which is a safety design that ensures a constant air supply for combustion and cooling. During the experiment it was discovered that a pressure drop across the burner in excess of 2 psi resulted in flame loss due to an excessive amount of air leading to blow out. The findings are inconclusive during the experiment in regards to the pressure losses at stages 12 – 16 because the HBC prototype never reached its intended operational pressures. However, it is assumed that the pressure drop between stage 12 and 16 will range from ½ psi to 2 psi.

Temperature effects and losses – The initial conditions for modeling this study are set at 15˚C (288.15 K). The current model can accept any initial values for temperature above zero Celsius and within the limits of practical conditions. The model was not tested for ambient temperatures below 0˚C. The effects of temperature loss within the HBC system were assumed to be accounted for within the efficiencies of the HBC components. Temperature losses at different stages were assumed to be negligible due to the use of insulation on the HBC. The maximum temperature of the burner is design not to exceed 732.2˚C (1350˚F) in a test facility. The model adopted this constraint during the simulation. There are alternative materials that can facilitate steady combustion of the burner at higher temperatures, however this improvement was not investigated as it is outside of the parameters for this thesis. The effects of dissociation of mixed fluid are only significant in mixed fluids above 1200 K [26, 27]. The HBC's thermodynamic

components do not operate above the specified temperature, for this reason all effects of dissociation in mixed fluids are ignored in this study.

Component performance – The efficiencies of all the components used within the HBC system are assumed based on common other industrial practices. The following are the assumed efficiencies:

- Compressor – 80%
- Turbine – 80%
- Electric generator – 95%
- Combustor chamber – 100% (assumed complete combustion of fuel)
- Heat exchanger – based on D.Q Kern proposed heat exchanger model [34]

The efficiency numbers can be improved, upon the availability of the HBC's experimental data. The accuracy of the simulation model is expected to improve with experimental efficiency values to realistically capture actual characteristics of the HBC components.

Simulation accuracy - Derivations and extrapolations performed in this study attempt to achieve an accuracy of 0.1%. During the iterations of the HBC model's temperature and pressure components, the acceptable variation between iterations is less than 1% temperature variation and less than 0.1% for pressure variation. Refining these tolerances further showed no significant change in the simulation results but only increases the time for computations. There are some exceptions to this general rule. The water spray module is adjusted to 2% declination from the desired result as 0.1% did not always achieve convergence during iterations.

Simulation results are shown in Figures 36 to 39 and Tables 3 and 4. The O – model shows that the addition of water into working fluid leads to increase in the efficiency of the biomass power cycle and reduces the TIT. Table 3 outlines the simulation's predicted performance values marked in green. The optimal fuel mass flow rate is determined at the proximity of cycle efficiency and system efficiency. The cycle efficiency always should exceed the system efficiency in order to have a thermodynamically valid simulation. Initial conditions that lead to

system efficiency being higher than cycle efficiency suggest that numerical simulation is venturing outside the thermodynamic boundary limits. Practically, this may mean that we introduce an excess amount of fuel into the simulation that in reality could not burn off or trigger elevated temperatures in the burners. Some boundary limitations were introduced in simulations but consideration was given to the simulation the freedom of movement across the range of operational conditions during iteration and to balance of the cycle. The O – model simulation shows that increasing the fuel injection rate above 0.00242 kg/sec creates an invalid solution in the simulation (system efficiency cannot be greater than cycle efficiency) which can indicate losses due to expected heat exchanger limitations. It is important to note that O – model is used outside its defined operational range that outline by Nelson and Sauer [27] and it relay on graphical extrapolation of the enthalpy and entropy past the define range.

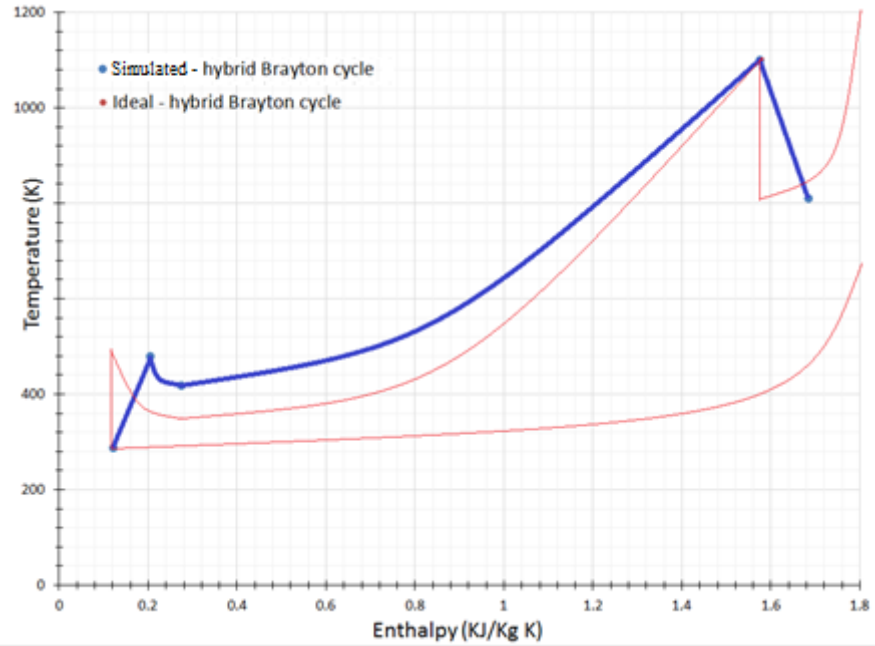


Figure 36 – S –T Graph for ideal and simulated Hybrid Brayton cycle values. Results are taken from CR 4.49, O model at 0.0042 kg/sec fuel injection and 0.0168 kg/sec water injection.

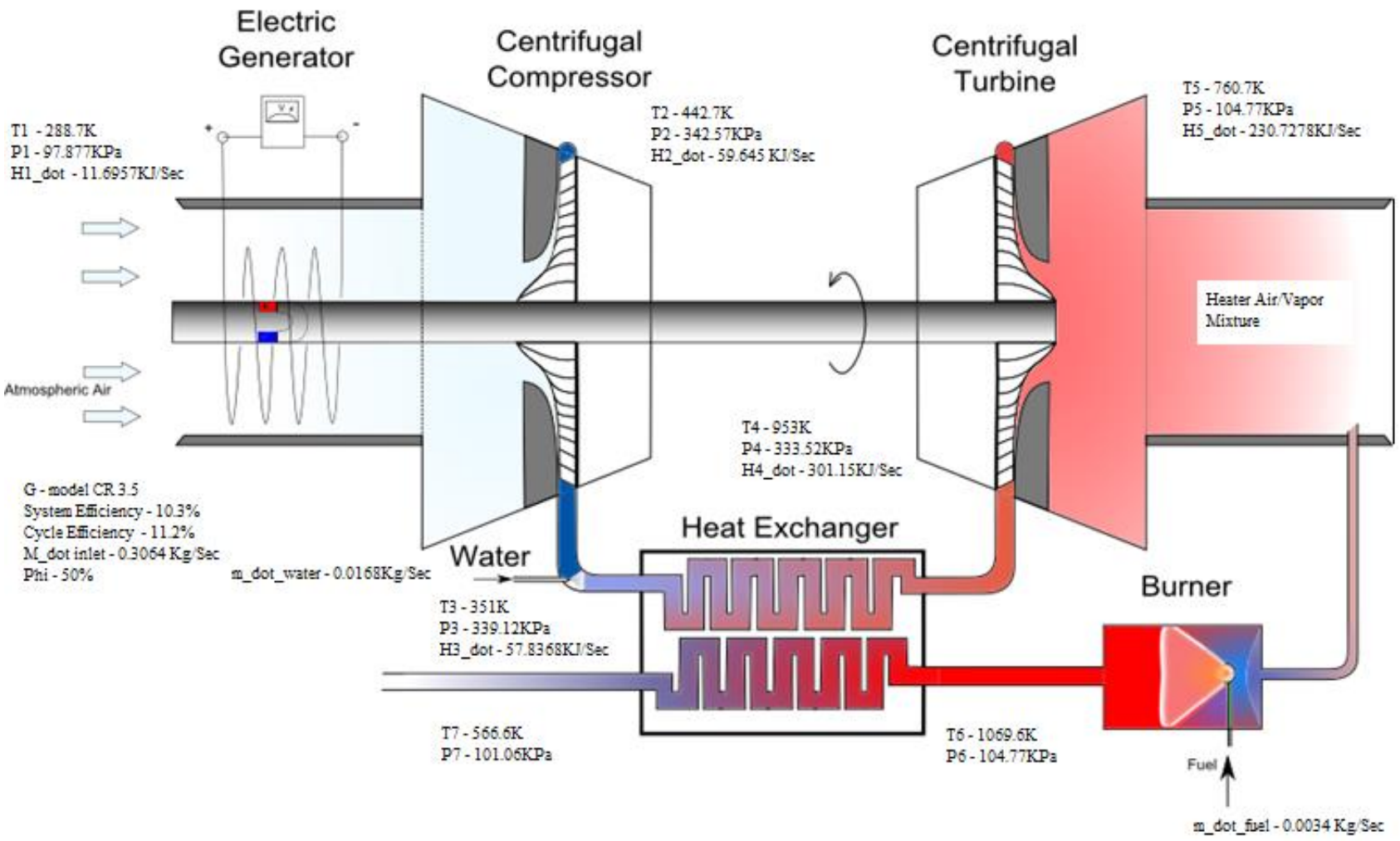


Figure 37 Overview of Hybrid Brayton cycle calculations in a G – model

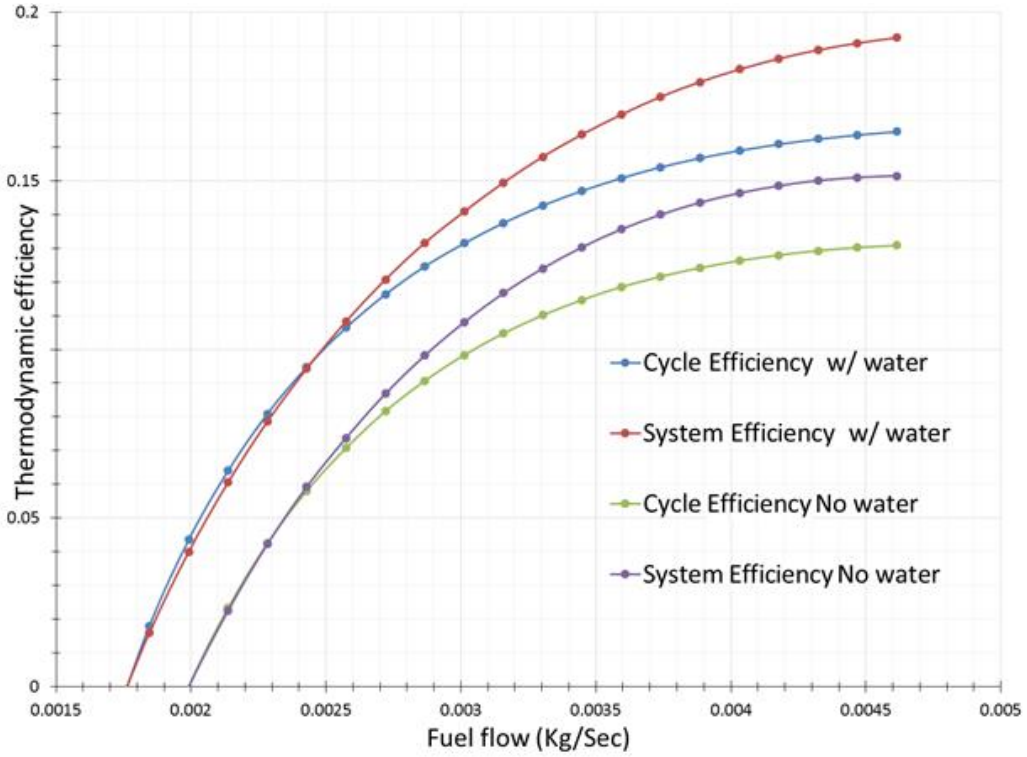


Figure 38 – Hybrid Brayton cycle performance in accordance with the O – Model at compression ratio 4.5. Corresponding fuel flow to crossing between cycle and system efficiency will suggest optimal fuel injection. No boundaries were set on combustion temperature.

Table 3 – O model simulation results using improved virial coefficient and compression ratio of 4.49

O - Simulation Model with improved Virial coefficent of Bww and Baw							
HBC system simulation performance with Water 0.0168Kg/Sec				HBC system simulation performance with No water			
Cycle efficenc	System ef	Fuel (Kg/Sec)	TIT (K)	Cycle efficiency (% fraction)	System efficiency (% fraction)	Fuel (Kg/Sec)	TIT (K)
(W/Q_HE)	(W/Qin)			(W/Q_HE)	(W/Qin)		
0.16461637	0.192468	0.004614331	1167.615456	0.130869509	0.151505037	0.004614331	1256.797
0.163627032	0.19088	0.004468614	1146.259796	0.130225082	0.151027631	0.004468614	1236.486
0.162399124	0.188822	0.004322898	1124.31456	0.129288519	0.150106312	0.004322898	1215.916
0.160877015	0.186256	0.004177181	1101.838946	0.127970716	0.148541462	0.004177181	1194.084
0.158983981	0.183089	0.004031465	1078.829052	0.126303117	0.146392115	0.004031465	1171.44
0.156749062	0.179359	0.003885748	1055.354959	0.124215494	0.14358227	0.003885748	1148.009
0.154081135	0.174972	0.003740032	1031.368374	0.121612166	0.139984839	0.003740032	1123.789
0.150811962	0.169666	0.003594315	1006.499637	0.118520485	0.135654301	0.003594315	1098.816
0.147045601	0.163787	0.003448599	981.8574385	0.114693016	0.130316007	0.003448599	1073.124
0.142699717	0.157158	0.003302882	956.8523649	0.110156501	0.124035379	0.003302882	1046.755
0.137510205	0.149481	0.003157165	931.2593426	0.104753043	0.116667547	0.003157165	1019.77
0.131477483	0.140865	0.003011449	905.4525404	0.09832345	0.108050659	0.003011449	991.7773
0.124591732	0.13159	0.002865732	880.7183951	0.090677007	0.098172424	0.002865732	963.8725
0.116313118	0.120724	0.002720016	854.7239091	0.081637402	0.086845813	0.002720016	935.204
0.106447154	0.108323	0.002574299	828.2429658	0.070738906	0.07374826	0.002574299	906.1143
0.094742766	0.094313	0.002428583	801.5379997	0.058044322	0.059249196	0.002428583	878.0177
0.080878604	0.078613	0.002282866	775.084705	0.042468085	0.042239718	0.002282866	848.4283
0.063981037	0.060553	0.00213715	748.5243458	0.023300166	0.02249058	0.00213715	818.4659
0.04350816	0.039962	0.001991433	721.8725289	-8.36E-05	-7.80E-05	0.001991433	788.589
0.017900643	0.015895	0.001845717	695.1442391	-0.029438341	-0.02642251	0.001845717	758.5841
-0.014434156	-0.01232	0.0017	668.0304679	-0.067025962	-0.057435575	0.0017	728.0912

Table 4 – G model simulation results using CR 4.0

HBC system simulation performance with No water					HBC system simulation performance with No water			
Cycle efficency (% fraction) (W/Q_HE)	System efficency (% fraction) (W/Qin)	Fuel (Kg/Sec)	TIT (K)	Water (Kg/Sec)	Cycle efficency (% fraction) (W/Q_HE)	System efficency (% fraction) (W/Qin)	Fuel (Kg/Sec)	TIT (K)
0.095939858	0.09977533	0.003364	909.2	0.016821	0.096293599	0.095430639	0.00303	981.8
0.10798178	0.116287698	0.00362	961.9	0.016821	0.090709566	0.087844755	0.00326	958
0.101676384	0.107497894	0.003502	936.1	0.016821	0.084485375	0.080889224	0.00315	940
0.097480697	0.101607698	0.003384	913.4	0.016821	0.078452769	0.073636017	0.00305	917.3
0.110024848	0.118854134	0.003628	964.8	0.016821	0.091901342	0.090043115	0.00327	964.4
0.101721245	0.107296494	0.003497	934	0.016821	0.085115201	0.081780171	0.00315	941.8
0.097507579	0.101565122	0.003366	912.4	0.016821	0.076592694	0.071909565	0.00303	915.6
0.104856253	0.112340783	0.003594	954.7	0.016821	0.089581239	0.086971451	0.00323	955.6
0.100186064	0.105622491	0.003449	928	0.016821	0.082043638	0.078166893	0.0031	932
0.092348385	0.095558839	0.003303	901.6	0.016821	0.092684173	0.091306237	0.0033	971.6
0.087114343	0.088109784	0.003157	872.4	0.016821	0.086932227	0.083699716	0.00316	943
0.07393369	0.072515392	0.003011	839.9	0.016821	0.075797905	0.070585166	0.00301	909.1
0.063965806	0.061639604	0.002866	815.4	0.016821	0.064555112	0.058460091	0.00287	880.4
0.053508756	0.049961984	0.00272	784.9	0.016821	0.052049637	0.045427411	0.00272	848
0.040587527	0.037432463	0.002574	766.9	0.016821	0.037192226	0.031471875	0.00257	821.1
0.026084351	0.023592846	0.002429	744.7	0.016821	0.015677396	0.012686425	0.00243	789.4
0.006181386	0.005400506	0.002283	717.2	0.016821	-0.002572963	-0.002027631	0.00228	769
-0.021961755	-0.018311898	0.002137	686.7	0.016821	-0.026704854	-0.020215907	0.00214	744.1
-0.054793552	-0.04329009	0.001991	655.2	0.016821	-0.063209428	-0.045048788	0.00199	713.3
-0.095648531	-0.071819786	0.001846	627.3	0.016821	-0.111867939	-0.073705535	0.00185	679.9

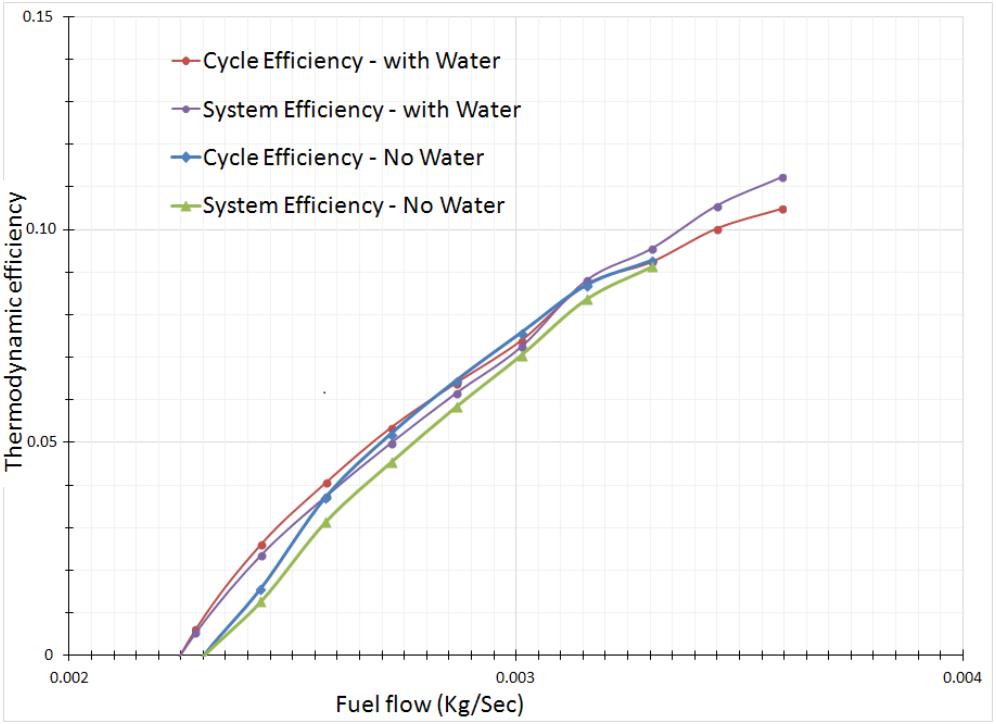


Figure 39 – G – model simulation results of Hybrid Brayton cycle and system efficiency at compression ratio of 4.0. Model used temperature limit of 1100K at the burner. The limit usage created reparative reading, as shown on Table 4, and these were removed off the graph, for clarity.

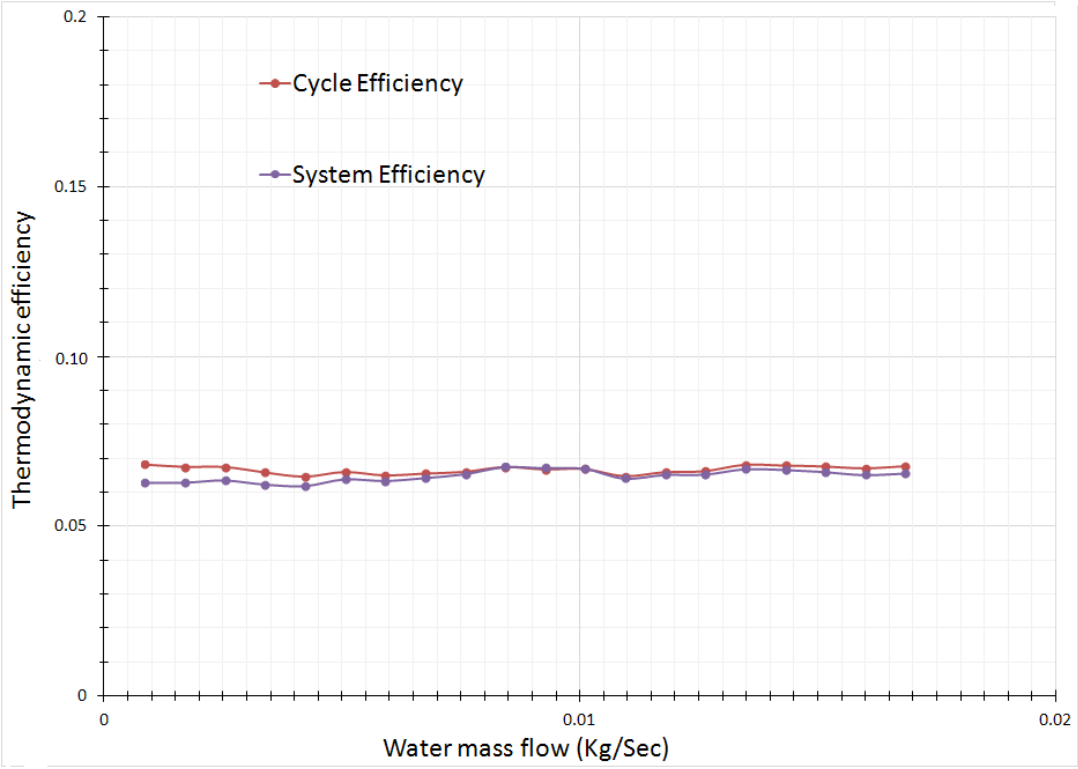


Figure 40 – G – model graph simulating the injection of water and its effect on efficiency of the cycle and system. Simulation used constant fuel flow of 0.00291kg/sec, and compression ratio of 4.0.

G – model does not show a consistent behaviour with that of an O – model. The effect of water addition into the HBC are consistent with the expected results. There is no expectation that solely increasing the mass flow rate leads to increase in efficiency as demonstrated in Figure 40. But rather addition of water allows the system to tolerate higher fuel consumption rates which leads to increase in efficiency as can be seen in Figure 39 and table 4.

The effect of compression ratio can be seen both on Figure 41 and 42. The limiting factor is the combustor outlet temperature that plays a pivotal role in making valid conclusion. This limit is not imposed when generating Figure 41 but is applied for Figure 42. Compression Ratio does not create a more effective system but rather, similar to effects of water addition, a higher CR allows the system to run at higher fuel combustion rates without major effects on the operational temperatures within the system. The increase CR may improve the thermodynamic efficiency of the turbine which is an assumed constant value in this analysis.

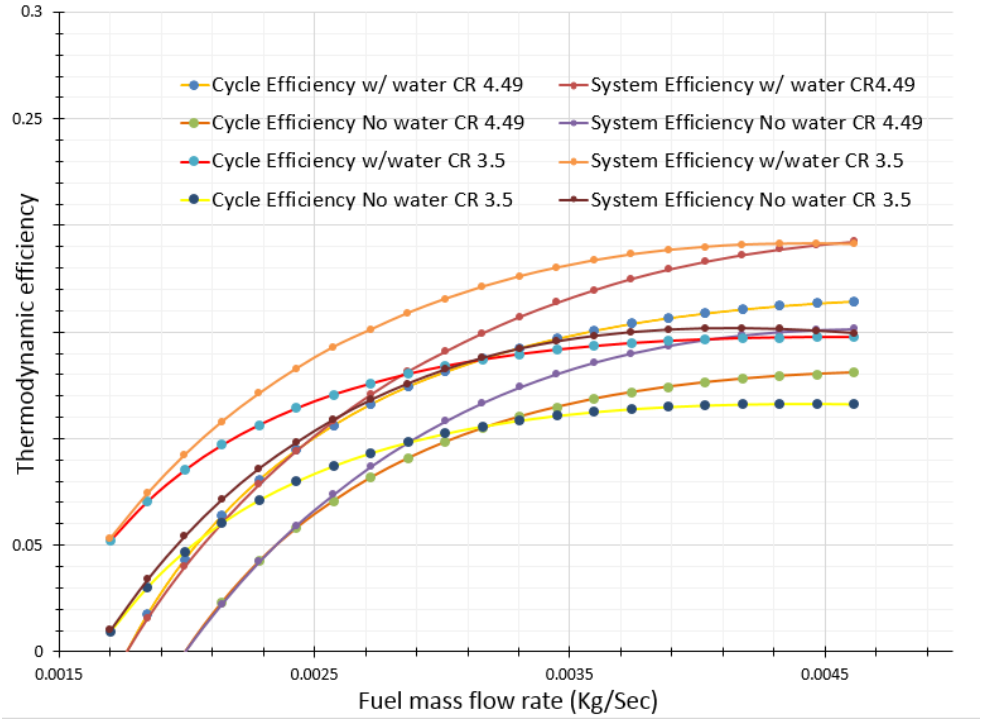


Figure 41 – O – model efficiency changes in Hybrid Brayton cycle with compression ratio change and presence of water injection. No combustor outlet temperature constrains used.

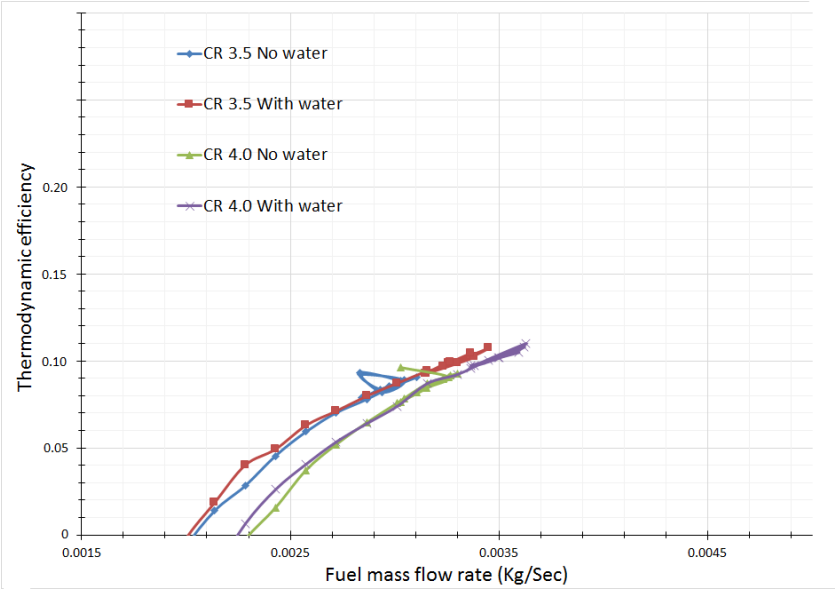


Figure 42 – G – model cycle efficiency representation at different compression ratios and fuel injection range. The simulation that was used to generate these curves was constrained by the combustor temperature of 1100K.

4.3 Deduction and recommendation

The G – model is expected to deliver more accurate and conservative results as compared to the O – model because it is designed to operate in the HBC's temperature and pressure range. O – model implementation required extrapolation past the confirmed range of the model. The graphs above also demonstrate the G – model is conservatively outlining the effect of water introduction into the system, while O – model shows an exaggerated effect of water on the system.

None of the proposed air/water models used in this study are experimentally validated at their operational range in this study. For this reason the exact efficiency of the HBC can only be predicted at this stage of the power system development. However, both simulations used in this study have demonstrated that addition of water into the Hybrid Brayton cycle will produce a potential for an increase in cycle and system efficiency and achieve turbine power output. With the commissioned model, the experimental results can be used to better assess the two simulation models presented in this study. It is expected that G-model will be confirmed as the more favourable model for simulation of thermodynamic processes like HBC. The HBC cycle addresses how to make biomass power system using a microturbine and indirectly fired.

From scientific publications it is clear that the study of Herrmann et al. recommends its formulation as the most accurate model for air/water simulation known currently to the scientific community. In addition, Nelson and Sauer air/water model is restrictive with its experimentally confirmed range.

Based on the information outline in this chapter, this study found that the Herrmann et al. simulation model is more appropriate for thermodynamic assessment of the HBC system or any other humid power cycle.

The process of commissioning a power cycle is lengthy in nature. This study brought HBC system closer to a fully functional power cycle. The first experimental unit used in this study showed that the HBC can be fully equipped in a 40 foot trailer, it can be fabricated at cost of about 4,000 to 5,000 $/kw, and controls are developed to operate this cycle. This unit has the

potential to produce 12 cent/kW-hr of electricity and approximately 3 cent/kW-hr of thermal energy assuming fuel is 15 MJ/kg Bone Dry at a cost of $40 per Bone Dry Tonne with HBC conservatively set at a 12% electrical system efficiency and 55% CHP efficiency.

5. Conclusion

This study investigated implementing the HBC as a viable and economically feasible solution for the small-scale power generation market. The technology for this application does exist today, and with additional commissioning trials a prototype can be operational. The learned lessons and tools developed in this study will be beneficial for other commissioning projects of HBC-like systems and for commercial implementation of the HBC system.

The current setup of the experimental facility is the first prototype of a HBC power cycle. It demonstrated some fundamental shortfalls with the chosen approach to the HBC test system configurations. In particular, the external pneumatic start up approach is found to be inappropriate for the C30 Capstone gas turbine. A C30 Capstone gas microturbine is equipped with air bearings, a design that is proven to function well with an electric start up. Reconfiguring that design for a pneumatic setup was found to be a challenging process and was not completely successful during this stage of the study. Excessive pressure loss across the air bearings due to design for thermal expansion, did not allow the system to engage these bearings and bring the system into a motion. Consequently this study recommends that the next attempt in the process of commissioning the HBC system, should be to consider implementing an electric start.

The Thermolib simulation, an off the shelf simulation product for thermodynamic cycles, did not produce the expected results for this study due to its idealistic mixing application used in the program. This study highlighted that the fundamental assumptions which were made in the Thermolib environment oversimplified all the critical phenomena that are essential for the proper simulation of the HBC cycle.

The two humidified air models that have been examined by this author are Nelson's model (original and with updated virial coefficient values) and Herrmann's model. Both delivered predictable results and can be used to simulate the HBC. Herrmann's simulation promises an accurate numerical solution to that of Nelson's model, with experimental backing to support these numbers. Although initial Herrmann's model implementation is challenging, accommodations are made in this study to ensure that Herrmann's model will perform

consistently to Nelson and Sauer's model to compare the two in the same environment. Results have shown that both Herrmann's and Nelson's models produce similar affect as an addition of water and an effect of CR. However Herrmann's model showed a more conservative pattern of numerical results for system/cycle efficiencies, a more favorable position during the commissioning phase. Therefore, based on these findings this studies recommends to adopt the Herrmann's study for future HBC simulations.

Future work on this project may include:

- Solve the technical problem of HBC. In particular, address the air bearing problem by using an electric start up protocol. Overcoming this obstacle will allow the system to reach a steady state.
- Confirm the recommended model results against the actual results to determine the extent of deviation and reasons for deviation.
- Modify the model parameters to reduce or eliminate deviation between the model and actual results.
- Optimise the layout of the HBC within the simulation environment in order to develop an economical production unit.
-

Additional research into HBC with outlined recommendation in this study may yield a functional unit in the next iterations of commissioning.

Bibliography

1. E. Brown and M. Mann. (2008, January). Initial Market Assessment for Small-Scale Biomass-Based CHP. National Renewable Energy Laboratory (NREL), Golden, CO, White Paper NREL/TP-640-42046 [Online]. Available: http://www.osti.gov/bridge/index.jsp [June 13,2012]

2. T. Levin and V. M. Thomas. (2011, May). Least-cost network evaluation of centralized and decentralized contributions to global electrification. Energy Policy, 2012 [Online], vol 41 pp. 286–302. Available: SciVerse ScienceDirect [14 June 2012]

3. World Alliance for Decentralized Energy, World Survey of Decentralized Energy 2006, WADE. Edinburgh, Scotland, UK [Online]. Available: www.localpower.org [14 June 2012]

4. D. Peterson and S. Haase,(2009, July) Market Assessment of Biomass Gasification and Combustion Technology for Small- and Medium-Scale Applications. National Renewable Energy Laboratory, Golden, Colorado [Online]. Available: http://www.osti.gov/bridge [14 June 2012]

5. Ministry of Natural Resources, Background to the Development of Ontario's Forest Fire Management Strategy. Ontraio Provincial Government, Toronto, Ontario [Online]. Available: http://www.mnr.gov.on.ca/ [14 July 2012]

6. P. Turgeon, P. Michel, P. Levallois, M. Archambault, A. Ravel, (2011, May) Fecal Contamination of Recreational Freshwaters: the Effect of Time-Independent Agroenvironmental Factors, SpringerNetherlands, Water Quality, Exposure and Health October 2011 [Online], vol 3, n 2, pp. 109-118. Available: SpringerLink [14 Jun 2012].

7. H. Steinfeld, P.Gerber, T. Wassenaar, V. Castel, M. Rosales, C. De Haan, Livestock's Long Shadow - Environmental Issues and Options, Food and Agriculture Organization of

the United Nations [Online], Rome, Italy, 2006. Available: http://www.fao.org/ [14 Jun 2012]

8. U.S. Energy Information Administration, Emissions of Greenhouse Gases in the United States 2009, Office of Integrated Analysis and Forecasting, U.S. Department of Energy ,Washington, DC, March 2011 [Online] Available: ftp://ftp.eia.doe.gov/environment/057309.pdf [21 Jun 2012]

9. International Energy Agency, Technology roadmap - Biofuels for transport. International Energy Agency, 9 rue de la Fédération Paris Cedex 15, France, 2011[Online]. Available: www.iea.org [21 Jun 2012]

10. M. Kirjavainen, K. Sipalā, T. Savola, M. Solamón and E. Alakangas (2004), Small scale biomass CHP technologies – Situation in Finland, Denmark and Sweden. OPET report 12, VTT Processes, Espoo, Finland [Online]. Available: http://www.opet-chp.net [21 Jun 2012]

11. M. Tampier, P.A. Beauchemin, D. Smith, E. Bibeau (2006, February), Identifying Enviromentally preferable uses for Biomass Recources - BC Bugwood: Economics, Technical Feasibility and GHG Implications of Seven Small to Medium Scale Technologies. North Vancouver, BC : Envirochem Services Inc, 2006 [Online]. Available: www.for.gov.bc.ca [21 Jun 2012]

12. C. Soares, Gas Turbines: A handbook for air, land and sea applications. Burlington, MA : Elsevier Academic Press, 2008.

13. M.P. Boyce, Gas Turbine Engineering Handbook 4th Ed. Waltham, MA : Elsevier Academic Press, 2012.

14. C.F. McDonald, Helium Turbomachinery Operating Experience from Gas Turbine Power Plants and Test Facilities, Applied Thermal Engineering, 2012, Vol. 44, pp. 108-181 [Online]. Available: http://www.sciencedirect.com. [15 Apr 2013]

15. C.W. Forberg, P.F. Peterson, H. Zhao, High-Temperature Liquid Fluoride-Salt Closed-Brayton Cycle Solar Power Towers, Journal of Solar Energy Engineering, May 2007, Vol. 129, pp. 141-146 [Online]. Available: DOI: 10.1115/1.2710245.

16. Canadian Renewable Fuel Association, Growing Beyond Oil Delivering Our Energy Future, Canadian Renewable Fuel Association, Ottawa, ON, 2010, pp. 7-8 [Online]. Available: http://www.greenfuels.org. [15 May 2013]

17. Turboden srl., Combined heat & Power units with split, InvisibleStudio, 2013 [Online]. Available: http://www.turboden.eu/en/products/products-chp.php. [13 May 2013]

18. BIOCAP Canada, An Information Guide on Pursuing Biomass Energy Opportunities and Technologies in British Columbia, Victoria, British Columbia, BC Ministry of Energy, Mines and Petroleum Resources, 2008 [Online]. Available: http://www.energyplan.gov.bc.ca [13 May 2013]

19. F. Dalili, Humidification in Evaporative Power Cycles, Doctoral Thesis, Energy Sciences, Dept. of Chem. Engr. and Tech., Royal Institute of Technology, Stockholm, Sweden, 2003.

20. M. Tampier, D. Smith, E. Bibeau, P.A. Beauchemin, STAGE 2 REPORT: Life-Cycle GHG Emission Reduction Benefits of Selected Feedstock-to-Product Threads, EnviroChem Services Inc., North Vancouver, BC, 2005.

21. J. Yan, L. Eidensten, Status and Perspective of Externally Fired Gas Turbines, Journal of Propulsion and Power, 2000, Vol 16, n 4, pp. 572-576 [Online]. Available: DOI:10.2514/2.5641

22. J. De Ruyck, F. Peters, S. Bram and G. Allard, Externally Fired Evaporative Gas Turbine Cycle for Small Scale Biomass CHP Production, ASME, IGTI, Vol 9, pp. 631-640, 1994.

23. F.F. Huang, Thermodynamic Study of the Overall Performance of an Air-Turbine Steam-Turbine Combined-Cycle Power Plant Based on First-Law as Well as Second-Law Analysis, , American Society of Mechanical Engineers, 1991. Available ISSN 04021215.

24. D. Smith, HBC System Excel Model, Private Communication, Winnipeg : s.n., 2009.

25. M. Horn, Humidification Processes in Gas Turbine Cycles. Department of Science, Lund University. Lund, Sweden : Division of Thermal Power Engineering, Faculty of Engineering., 2005.

26. S. Herrmann, H.-J Kretzschmar, V. Teske, E. Vogel, P. Ulbig, R. Span, D.P. Gatley, Determination of Thermodynamic and Transport Properties of Humid Air for Power Cycle Calculations, Braunschweig, Germany : PBT, 2009.

27. H.F. Nelson, H.J. Sauer, Formation of High Temperature Properties for Hot Air, March 2002, HVAC & R Research, Vol. 8, pp. 311-334.

28. V.G. Beketov, V.A. Rabinovich, M.D. Rogovin, Calculation of the Thermodynamic Properties of Nonpolar Wet Gases at Elevated Pressures, 1995, Journal of Engineering Physics and Thermophysics, Vol 68, n 6, pp. 709-717.

29. E.W. Lemmon, R.T. Jacobsen, S.G. Penoncello and D.G. Friend, Thermodynamic Properties of Air and Mixtures on Nitrogen, Argon and Oxygen from 60 to 2000 K at Pressures to 2000MPa, July 2000, Journal of Physics and Chemical Reference Data, Vol 29, No. 3, pp. 331-385.

30. B.F. Kolanoswski, Guide to Microturbines. Lilburn, GA, The Fairmont Press Inc., 2004.

31. A. Traverso, A.F. Massardo, R. Scarpellini, Externally Fired Micro-Gas Turbine: Modeling and Experimental Performance, 2006, Applied Thermal Engineering, Vol. 26, pp. 1935-1941.

32. F.P. Incropera, D.P. DeWatt, T.L. Bergman, A.S. Lavine, Fundamental of Heat and Mass Transfer, 6th Ed. Hoboken, NJ : John Wiley & Sons, 2006.

33. A.P. Fraas, Heat Exchanger Design, 2nd Ed. Hoboken, NJ : John Wiley & Sons, 1989.

34. D.Q. Kern, Process Heat Transfer, McGrow Hill Book Company, 1950.

35. P.E. Santangero, P. Tartarini, Fire Control and Suppression by Water-Mist System, 2010, The Open Thermodynamic Journal, Vol. 4, pp. 167-184.

36. Bete Fog Nozzle Inc. "Drop Size Average Definition", Personal Communication, 2012.

37. EUtech Scientific Engineering. Thermolib - Thermodynamic Systems Library, Release 4.2 User Manual. Aachen, Germany, 2009.

38. Y. Ohkubo, Outlook on Gas Turbine, 2005, R&D Review of Toyota CRDL, Vol. 41 [Online]. Available: http://www.tytlabs.com [13 May 2013]

Appendix A – Equations and coefficients for moist air by Nelson and Sauer

The Nelson and Sauer's study assumes the air composition to be 78.12% Nitrogen, 20.96% Oxygen, and 0.92% Argon.

Nelson and Sauer treat moist air as a mixture of two real gases (air and water vapour) and utilise the P-ν-T equation as described by the virial equation of state:

$$Z = \frac{P\bar{v}}{\bar{R}T} = 1 + \frac{B_m}{\bar{v}} + \frac{C_m}{\bar{v}^2} + \cdots \tag{A1}$$

Z is a compressibility factor for moist air, where Z=1 corresponds to ideal gas.

$$B_m = x_a^2 B_{aa} + 2x_a x_w + x_w^2 B_{ww} \tag{A2}$$

$$C_m = x_a^3 C_{aaa} + 3x_a^2 x_w C_{aaw} + 3x_w^2 x_a C_{aww} + x_w^3 C_{www} \tag{A3}$$

The P, T, x_a, x_w, B_m and C_m values can be used in equation #1 to determine the molar volume, molar enthalpy, and molar entropy of the moist air.

Virial coefficient

$$B_{aa} = 31.831763 - \frac{719.51195}{T} - \frac{6{,}538{,}137.0}{T^2} + \frac{1.5929828 \times 10^9}{T^3} - \frac{2.5588842 \times 10^{11}}{T^4} + \frac{2.2300382 \times 10^{13}}{T^5} - \frac{8.2793465 \times 10^{14}}{T^6} \tag{A4}$$

$$B_{ww} = -4965.8164 + \frac{1.81918860 \times 10^7}{T} - \frac{2.92013995058 \times 10^{10}}{T^2} + \frac{2.7032989 \times 10^{13}}{T^3} - \frac{1.6045262 \times 10^{16}}{T^4} + \frac{6.3750397 \times 10^{18}}{T^5} - \frac{1.7206027 \times 10^{21}}{T^6} + \frac{3.1222306 \times 10^{23}}{T^7} - \frac{3.6643847 \times 10^{27}}{T^8} + \frac{2.5256562 \times 10^{27}}{T^9} - \frac{7.929348 \times 10^{28}}{T^{10}} \tag{A5}$$

$$B_{aw} = 32.366097 - \frac{14{,}113.8}{T} - \frac{1{,}244{,}535}{T^2} - \frac{0.2348789\times10^{10}}{T^4} \quad \text{(A6)}$$

$$C_{aaa} = 1297.5378 + \frac{46{,}021.328}{T} + \frac{40{,}813{,}154.0}{T^2} - \frac{3.202391\times10^{9}}{T^3} + \frac{2.2964785\times10^{11}}{T^4} - \quad \text{(A7)}$$

$$\frac{5.3683467\times10^{12}}{T^5} - \frac{2.1183915\times10^{14}}{T^6}$$

$$C_{aaw} = 4823.737 + \frac{105{,}678}{T} - \frac{6.56394\times10^{7}}{T^2} + \frac{2.94442\times10^{10}}{T^3} - \frac{3.19317\times10^{12}}{T^4} \quad \text{(A8)}$$

$$C_{aww} = -10^{6} exp\left[-10.72887 + \frac{3478.04}{T} - \frac{383{,}383}{T^2} + \frac{33{,}406{,}000}{T^3}\right] \quad \text{(A9)}$$

$$C_{www} = -10^{6} exp[-6.566276606 + 0.3894679516T - 0.00342810205 37T^2 + 1.333924918\times10^{-5}T^3 - 2.726404078\times10^{-8}T^4 + 2.839369136\times10^{-11}T^5 - 1.189114330\times10^{-14}T^6] \quad \text{(A10)}$$

Water properties

$$\frac{\bar{g}_{ig}\,(P,T)}{\bar{R}T} = \gamma^{\circ} = \ln\pi + \sum_{i=1}^{9} n_i^{\circ}\tau^{J_i^{\circ}} \quad \text{(A11)}$$

$$\bar{v}_{ig} = \bar{R}T/P \quad \text{(A12)}$$

$$\bar{h}_{ig^{\circ}} = \bar{R}T\tau\sum_{i=1}^{9} n_i^{\circ} J_i^{\circ}\tau^{J_i^{\circ}-1} \quad \text{(A13)}$$

$$\bar{s}_{ig^{\circ}} = \bar{R}\left[\tau\sum_{i=1}^{9} n_{i^{\circ}} J_i^{\circ}\tau^{J_i^{\circ}-1} - \gamma^{\circ}\right] \quad \text{(A14)}$$

The table below is from Wagner and Kruse (1998) and Parry et al.(2000)

n	-9.6927687	10.086656	-0.0056088	0.07145274	-0.407105	1.42408192	-4.3839511	-0.2840863	0.02126846
J	0	1	-5	-4	-3	-2	-1	2	3

Water vapour saturation pressure

$233.15 \le T \le 273.15K$ $(-40 \le T \le 0^{\circ}\ C)$

$$P_S = -1{,}922{,}159.5948 + 40{,}288.865238T - 388.50127537T^2 + 1.425383278154T^3 - 3.0089728994 \times 10^{-3}T^4 + 2.5482607835 \times 10^{-6}T^5 \quad \text{(A15)}$$

$273.15 \leq T \leq 647.10K$ $(0 \leq T \leq 374^\circ C)$

$$P_S = P^{***}\left[\frac{2C}{B + \sqrt{B^2 - 4AC}}\right]^4 \quad \text{(A16)}$$

$$Q = \frac{T}{T^{***}} - \frac{0.238555576}{\frac{T}{T^{***}} - 650.175348} \quad \text{(A17)}$$

$$A = Q^2 + 1167.05215Q - 724{,}213.167 \quad \text{(A18)}$$

$$B = -17.0738469Q^2 + 12{,}020.8247Q - 3{,}232{,}555.03 \quad \text{(A19)}$$

$$C = 14.9151086Q^2 - 4823.26574Q + 405{,}113.405 \quad \text{(A20)}$$

Where $P^{***} = 1.0 \times 10^6 Pa$, $T^{***} = 1K$

Air properties

$$\bar{h}^{\circ}_{ig,a} = 28.921316(T - T_\circ) + \frac{2.5861872 \times 10^{-3}(T^2 - T_\circ^2)}{2} - \frac{1.9010204 \times 10^{-5}(T^3 - T_\circ^3)}{3} + \frac{5.1208717 \times 10^{-8}(T^4 - T_\circ^4)}{4} - \frac{3.2775941 \times 10^{-11}(T^5 - T_\circ^5)}{5} + 8.0457 \quad \text{(A21)}$$

$$\bar{s}^{\circ}_{ig,a} = 28.921316ln\left(\frac{T}{T^\circ}\right) + 2.5861872 \times 10^{-3}(T - T_\circ) - \frac{1.9010204 \times 10^{-5}(T^2 - T_\circ^2)}{2} + \frac{5.1208717 \times 10^{-8}(T^3 - T_\circ^3)}{3} - \frac{3.2775941 \times 10^{-11}(T^4 - T_\circ^4)}{4} + 0.02393 \quad \text{(A22)}$$

Thermodynamic properties

$$\bar{h}_i = \bar{h}^{\circ}_{ig,i} + \bar{R}T\left[\left(B_{ii} - T\frac{dB_{ii}}{dT}\right)\frac{1}{\bar{v}_i} + \left(C_{iii} - \frac{1}{2}T\frac{dC_{iii}}{dT}\right)\frac{1}{\bar{v}_i^2}\right] \quad \text{(A23)}$$

$$\bar{h}_i = \bar{h}^{\circ}_{ig,i} + \bar{R}T\left[\left(B_{ii} - T\frac{dB_{ii}}{dT}\right)\frac{1}{\bar{v}_i} + \left(C_{iii} - \frac{1}{2}T\frac{dC_{iii}}{dT}\right)\frac{1}{\bar{v}_i^2}\right] \quad \text{(A24)}$$

$$\bar{h}_m = x_a \bar{h}^{\circ}_{ig,a} + x_w \bar{h}^{\circ}_{ig,w} + \bar{R}T\left[\left(B_m - T\frac{dB_m}{dT}\right)\frac{1}{\bar{v}_m} + \left(C_m - \frac{1}{2}T\frac{dC_m}{dT}\right)\frac{1}{\bar{v}_m^2}\right] \quad \text{(A25)}$$

$$\bar{s}_i = \bar{s}^{\circ}_{ig,i} + \bar{R}\left\{-ln\frac{P_i}{P^{\circ}} + ln\frac{P_i v_i}{\bar{R}T} - \left[\left(B_{ii} + T\frac{dB_{ii}}{dT}\right)\frac{1}{\bar{v}_i} + \left(C_{iii} + T\frac{dC_{iii}}{dT}\right)\frac{1}{2\bar{v}_i^2}\right]\right\} \quad \text{(A26)}$$

$$\bar{s}_w = s^{\circ}_{ig,w} + \bar{R}\left\{-ln\frac{P}{P^{\circ}} + ln\frac{P\bar{v}_a}{\bar{R}T}\right\} - \left[\left(B_{ww} + T\frac{dB_{ww}}{dT}\right)\frac{1}{\bar{v}_w} + \left(C_{www} + T\frac{dC_{www}}{dT}\right)\frac{1}{2\bar{v}_w^2}\right] \quad \text{(A27)}$$

$$\bar{s}_a = s^{\circ}_{ig,a} + \bar{R}\left\{-ln\frac{P}{P^{\circ}} + ln\frac{P\bar{v}_a}{\bar{R}T}\right\} - \left[\left(B_{aa} + T_s\frac{dB_{aa}}{dT}\right)\frac{1}{\bar{v}_a} + \left(C_{aaa} + T_s\frac{dC_{aaa}}{dT}\right)\frac{1}{2\bar{v}_a^2}\right] \quad \text{(A28)}$$

$$\bar{s}_m = x_a \bar{s}^{\circ}_{ig,a} + x_w \bar{s}^{\circ}_{ig,w} + \bar{R}\left\{-ln\frac{P}{P^{\circ}} + x_a ln\frac{P\bar{v}_m}{x_a\bar{R}T} + x_w ln\frac{P\bar{v}_m}{x_m\bar{R}T} - \left[\left(B_m - T\frac{dB_m}{dT}\right)\frac{1}{\bar{v}_m} + \left(C_m + T\frac{dC_m}{dT}\right)\frac{1}{2\bar{v}_m^2}\right]\right\} \quad \text{(A29)}$$

Appendix B – Hybrid Brayton cycle test facility safe operation and troubleshooting

B.1 Safety

The HBC test facility has several identified safety aspects. All personnel present on site must follow safety instructions in effort to prevent accidental injury.

- Noise – During the operation of HBC all present on site must wear approved hearing protection. Hearing protection in the control room, with door to the machine portion closed, is not required.

- Debris – The high air flow nature of the HBC can create high velocity debris. During the operation of the HBC all present must wear approved protective eyewear. Eye protection in the control room, with door to the machine portion closed, is not required.

- Hot surface – Most of the HBC is insulated with multi-layer insulation for thermal efficiency and safety of the operators. However, some surfaces are remain exposed and can reach several hundred degrees Celsius. During operation consider all exposed stainless steel surfaces of the HBC as hot, use thermal gloves to prove otherwise.

- Electricity – There are high voltage lines and delicate instrumentation that is used for the operation of HBC. Never work on any electric component until proven that it is safe to do so. Never work on electric components, when system is running.

B.2 Start-up controls

The HBC system has two control elements for this experiment.

The fuel flow control is achieved via burner control box that corresponds to the set temperature limits. The temperature controller sets the operational burner output temperature, where the

system will try to adjust the flow to reach that set temperature goal. The high temperature limit controls the upper limit of burner output. Exceeding the set limit will result in automatic burner shutdown (a reset on the dial must be done by pressing the reset button on the control panel). Temperature on this dial should never exceed 1350 F. See Figure B.1.

Figure B.1: Control panel found in Hybrid Brayton cycle experimental facility

The second control is manual valves that control the external compressed air flow for start-up through the system, shown in Figure B.2.

Figure B.2: Air control valves for external startup/shutdown

The left valve controls the flow for the start-up of the turbine, while the right links air direct into burner. Note that the burner requires ½ psi pressure drop in order to unlock the interlock safety (white indicator on burner control box). Minimal airflow must be passing across burner during the ignition.

Electric power load – The effect of power load shown in Figure B.3 are unknown at this stage of the experiment. Prior to starting the test:

- Test relays (controls are in top right corner of the control room)
- Plug in the heaters, to start their fans

Figure B.3: Electric load control panel.

Expected results: The HBC device to successfully demonstrate steady combustion at ~ 900°F with ~60 SLPM gas flow and achieved maximum flow rates of ~106 SLPM and temperatures of ~1100°F

Trouble shooting**:** Ensure all emergency stop buttons are disengaged.

B.3 Combustor

Troubleshoot the burner itself to make sure that spark generator is working and delivers power to the spark rod (use high voltage proximity indicator). Check if flame is present by removing ultra violet flame sensor and looking down the tube. Use glass to cover the tube to prevent air flow into eyes. The burner control system will attempt to light up a flame for 10 seconds, during that time operator can see if flame is present. The burner control will also indicate flame presence (Figures B.4 and B.5) during lighting and steady burn. If gas sparking does not result in steady combustion within ten seconds of the start-up, operator should attempt again with reduction in the Air flow during the light up process. Keep reducing airflow to minimum, just low enough to prove the interlock pressure sensor. Figure B.6 controls minimum air flow achieved when right air flow meter's gage is barely reaching green zone.

Figure B.4: Ultraviolet sensor (left) and sparker (right)

Figure B.5: Burner control module

Figure B.6: Air/water separators with air flow gages

B.4 Control box

Next move to the control box, if wiring problem suspected, a schematics of Maxon Wiring is present in the lab with user manual for Maxon burner, shown in Figure B.1. Note, the actual system that is used for HBC is simplified and for these reasons amendments were made in the actual control box. The current changes to electronic schematic should reflect accurately the actual electronics of the system.

The burner control consists of two timer relays, three contact relays and a burner control module, as shown in Figures B.7 and B.8. When the system pressure safety interlock is proven the control panel can start with the start-up cycle (system requires 0.5 psi pressure drop across the burner). The system will start with the purge cycle that is controlled by relay timer (TD 107). The purge cycle must last at minimum 30 seconds in order to guarantee no excessive accumulation of propane gas occurs around the spark rod. With the completion of the burner purge, the system will attempt to light up the burner. During this step the burner control module is engaged in a 10 second start up protocol. During that same time period, the second time relay (TD133) is engaged for 8 seconds to override some of the safety interlocks and provide power to the gas line despite lack of flame. The three contact relays are used to interrupt gas flow. If Stop

button is pressed, link all Stop buttons in the trailer and engage gas valves on the gas train, shown in Figure B.7.

Figure B.7: Gas train left to right: flow meter, pressure reducer/mechanical overpressure blow valve, electric high pressure gage with cut off maximum pressure, shut off valve, pressure reducer and regulator, low pressure gage w/ max cut off pressure, butterfly gas valve with digital controller

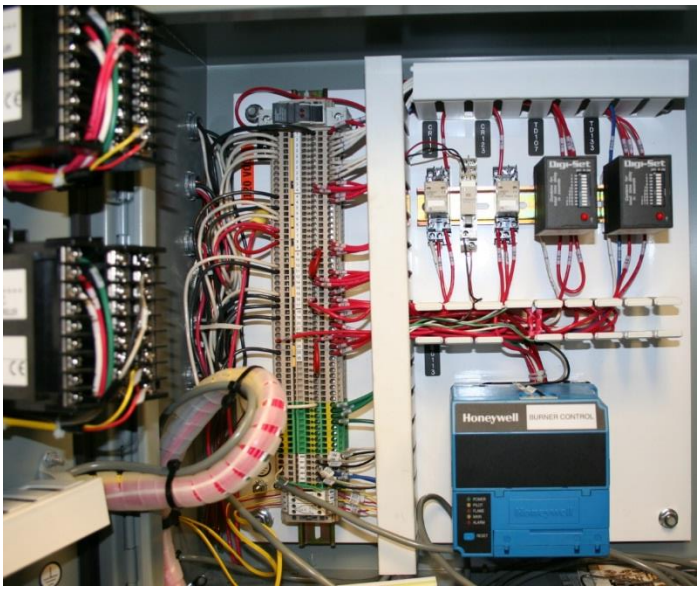


Figure B.8: Inside the burner control box

During the light up, there is a back pressure created that may trip one of the pressure safety sensors on the gas line. This situation may result in restart of the process without reset after 8 to 10 seconds of the light up process. To rectify this situation, adjustment to the pressure setting needs to be made before attempting again.

B.5 Control and operation of the system

In order to achieve sustainable performance from the burner, the flame from the burner must be optimal. This can be determined by the feedback from the ultraviolet flame sensor. Operator must place voltmeter on the outlets on the bottom right corner of the burner control module. Sensor provides 0-5.0 V feedback to the system that can be measured at the control module. The key for achieving steady combustions is to provide the right mixture of air and fuel and to maintain a steady signal from the sensor above 1.0 V. Fluctuating signal suggests that flame is not stable and system will shut down if signal of 0 V is read from the flame sensor for more than 3 sec.

B.6 Start Up Compressor

It is vital to conduct regular inspections on the compressor as some indicators are not functioning properly. The compressor unit cannot operate more than 3 hours without inspecting fuel levels, failure to do so may result in running out of fuel. At every refueling, oil check and antifreeze check must be done.

Figure B.9: External Compressor to trailer airline connection

B.7 Shut down

After shutting down the burner from the control panel, the compressor must run for 10 to 15 minutes before turning the air flow off, and further 5 more minutes at idle to cool down the compressor before shutting it off.

Appendix C – Hybrid Brayton cycle MATLab models' code

HBC SIMULATION - Humidity Variation Test O-Model

```
%%
F_S=1;% F_S - Fail safe controller, design to complete the program if non
%convergence is occurs.

%% Inlet Conditions
% Air

T_amb = 288.70556; % [K]
P_amb = 101325; %[Pa]
m_dot = (2432.06.*0.45359237)/3600; % [kg/s]
Phi = 0.5; % percent humidity

% water
m_dot_water =0;%(133.5.*0.45359237)/3600;%[kg/s]
T_w = 20+273.15; %[K]

%Fuel

T_Fuel = 290;%[K]
m_dot_fuel = (23.13.*0.45359237/3600); %[kg/s]

%% Exturnal Loop
Phi_ind = 0.0000001; % Phi indicator
m=int8(1);

T= zeros(20,7);
P= zeros(20,7);

% T_sat = zeros(20,7);

Sys_eff =zeros(20,1);
Cyc_eff = zeros(20,1);
fuel_Sys= zeros(20,1);
h_m_m = zeros(20,7);
percent_mass_conserv = zeros(20,1);
percent_energy_conserv =zeros(20,1);

while (m<22)

   if (F_S==1)% F_S - Fail safe controller
    %% Loop
    % initial conditions

    P6 = P_amb;
    T6_test = 1005.22;
    m_dot_fuel1=(m_dot_fuel-m_dot_fuel.*(0.05*(double(m)-1)))+0.0017;
    %0.0017+ is an approximate amount of fuel required to have positive
    %efficiency of the system w/ m_dot_water =(133.5.*0.45359237)
    %/3600;%[kg/s].
    if (m_dot_fuel1==0)
```

```
        m_dot_fuel1=0.00001;
    end
    m_dot6_CO2 = 0.01.*0.45359237/3600;
    m_dot6_N2= m_dot.*0.78;
    m_dot6_H2O =m_dot_water;
    m_dot6_O2 =m_dot.*0.22;

    T_calc= zeros(6,7);
    P_calc= zeros(6,7);
    T6 = T6_test;
    test = int8(0);
    test2 = int8(0);
    n =int8(1);

    % argument
    while (test==0)

        %% Compressor
        [P1,T1,h_m_m1, ~, ~,~,~,~, ...
            ~,~,P2,T2,h_m_m2, ~,~,Comp_Work_In,m_dot2,...
            m_dot_w2, m_dot_a2, x_a2, x_w2, M_m,F_S] = ...
            Compressor(T_amb,P_amb, Phi,m_dot,F_S);

        %% Spray #1
        [P3,T3,h_m_m3, ~, ~,m_dot3,...
            m_dot_w3, m_dot_a3, x_a3, x_w3, M_m3,F_S] = ...
            Spray1(T2,P2, m_dot_water,m_dot2,T_w,x_a2,x_w2,m_dot_a2,...
            m_dot_w2,F_S);

        %% Heat Exchanger (HE)

        % pressure drop due to pipes.
        P3_HE =((P3.*0.000145037738)-0.5)/0.000145037738;

        [T4,P4,T7,P7,Q_HE,F_S] = ...
            Heat_exchanger(P3_HE,T3,P6,T6, m_dot6_H2O,m_dot6_CO2,...
            m_dot6_N2,m_dot6_O2,m_dot_w3,m_dot_a3,F_S);

        %% Turbine

        [~, ~, ~,~,~,~,~,~, ~,P5,T5,h_m_m5, ~, ~,Turbine_Work_Out,F_S] =
Turbine(T4,P4,...
            m_dot_a3,m_dot_w3,P_amb,F_S);

        %% Combustor


[T6,P6,m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2,Q_total,Q_created,F_S] = ...
            Combustor(T_amb,P5,T5,m_dot_w3,m_dot_a3,T_Fuel,m_dot_fuel1,F_S);



            if (test2==1)
                test =1;
```

```
            P_calc(1,:) = [NaN; NaN; NaN; NaN; NaN; NaN; NaN];
            T_calc(1,:) = [NaN; NaN; NaN; NaN; NaN; NaN; NaN];


        else

            n=n+1;

            P_calc(n,:) = [P1; P2; P3_HE; P4; P5; P6; P7];
            T_calc(n,:) = [T1; T2; T3; T4; T5; T6; T7];

            P_diff = (P_calc(n,:)-P_calc((n-1),:))./(P_calc(n,:));
            T_diff = (T_calc(n,:)-T_calc((n-1),:))./(T_calc(n,:));
            chk = double((ones(1,7).*0.001));
            % Test for tolerance of 0.1%
            if (abs(P_diff) < chk)
                if (abs(T_diff) < chk)
                test2=1;% test2 confirms that tolerance is achieved.
                end
            end

    end

end

% Conservation tests function


 F_S = ConfirmationofLaws(m_dot2,m_dot3, m_dot_water, ...
   m_dot_fuel1, m_dot6_H2O, m_dot6_CO2, m_dot6_N2, m_dot6_O2);

if (F_S==0)
    m=30;
else

% Conservation of energy test

    P(m,:) = P_calc(n,:);
    T(m,:) = T_calc(n,:);

    h_m_m(m,:)= [ (h_m_m1.*m_dot); (h_m_m2.*m_dot2); ...
        (h_m_m3.*m_dot2);0;(h_m_m5.*m_dot2);0;0 ];% [KJ/sec]



    Sys_eff(m,1) = (-(Comp_Work_In.*1.05)-(0.95.*Turbine_Work_Out))...
        /Q_total;
    Cyc_eff (m,1)= (-(Comp_Work_In.*1.05)-(0.95.*Turbine_Work_Out))...
        /Q_HE;
    fuel_Sys(m,1) = m_dot_fuel1;
```

```
    end
        m=m+1;
   end
end
Sys_eff_plt = Sys_eff.*100;
Cyc_eff_plt = Cyc_eff.*100;
figure
plot(fuel_Sys(:,1),Sys_eff_plt,'--ok');
title('Fuel Injection Variation effects on System`s Efficency');
xlabel('Fuel Injection Rate [Kg/s]');
ylabel('System Efficency');
figure
plot(fuel_Sys(:,1),Cyc_eff_plt,'--ok');
title('Fuel Injection Variation affects on Cycle`s Efficency')
xlabel('Fuel Injection Rate [Kg/s]');
ylabel('Cycle Efficency');
clear P1 P2 P3 P3_HE P4 P5 P6 P7;
   clear T1 T2 T3 T4 T5 T6 T7;
```

Compressor

```
% Cycle stage 1-2 w/ Inlet temperature drop.

function [P1,T1,h_m_m1, s_m_m1, v_m_m1,P2_iso,T2_iso,h_m_m2_iso, ...
    s_m_m2_iso, v_m_m2_iso,P2,T2,h_m_m2, s_m_m2, v_m_m2,Work_dot,m_dot2,...
    m_dot_w2, m_dot_a2, x_a2, x_w2, M_m,F_S] = ...
    Compressor(T_amb,P_amb, Phi,m_dot,F_S)%[K,Pa,Humidity,Kg/sec]

clear v_m1 h_m1 s_m1 v_m2_iso h_m2_iso s_m2_iso v_m2 h_m2 s_m2;
if (F_S==1)% F_S - Fail safe controller
%% Definitions
P1 = ((P_amb.*0.000145037738)-0.5)/0.000145037738; %1 Pa =0.000145037738 Psi
                        %0.5 PSIG inlet pressure drop due to filter.
T1 = T_amb;
eff = 0.8; % Comressor thermal efficiency
[P_s] = Water_Vapor_Saturation ('P_s@T',T1);

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

%% Initial Conditions
MR = M_w/M_a; % Molar Ratio

P_v = Phi.*P_s;%[Pa]

MF = MR.*P_v/(P1-P_v); % Molar Fraction n_w/n_a
m_dot_a2 = m_dot/(MF/MR+1);%[Kg/sec]
m_dot_w2 = m_dot - m_dot_a2;%[Kg/sec]
m_dot2 = m_dot_w2 + m_dot_a2; %[Kg/sec] - Conservation of mass.

x_w2=(MF/MR)/((MF/MR)+1);
x_a2=1/((MF/MR)+1);
M_m=x_a2.*M_a+x_w2.*M_w;%[Kg/Kmol]

%% Thermodynamics @ Compressor Inlet
x_a1 = x_a2;
x_w1 = x_w2;
[v_m1, h_m1, s_m1]=Thermodynamic_Properties (T1, P1, x_a1, x_w1);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]

v_m_m1 = v_m1/M_m; %[m^3/Kg]
h_m_m1 = h_m1/M_m; %[KJ/Kg]
s_m_m1 = s_m1/M_m; %[KJ/Kg/K]

P2 = P1*4.42; % Pressure Ratio @ 4.49
%% Thermodynamics @ Compressor Outlet - Isotropic

test = 1;
diff1=-inf;
T2_iso = T1;
n=int32(0);
clear diff;
while (test==1)
```

```
    [v_m2_iso, h_m2_iso, s_m2_iso]=Thermodynamic_Properties (T2_iso, P2,...
        x_a2, x_w2);
    diff = s_m1-s_m2_iso;
    if (isreal(diff)==0)
         T2_iso=T2_iso+121;

    elseif (diff<-0.05)
            T2_iso=T2_iso-71;

    elseif ((abs(diff)>50))
            T2_iso=T2_iso+40;

    elseif ((abs(diff)>10))
            T2_iso=T2_iso+30;

    elseif (abs(diff)>1)
            T2_iso=T2_iso+15;

    elseif (abs(diff)>0.1)
            T2_iso=T2_iso+1;

    elseif (abs(diff)>0.05)
            T2_iso=T2_iso+0.1;

    else
            test =0;

    end
        n=n+1;
        if (n ==3000)
            test =0;
            F_S=0;
            msgbox('Did not converge','Compressor - Error');
        end

end


P2_iso = P2;%[Pa]
v_m_m2_iso = v_m2_iso/M_m; %[m^3/kg]
h_m_m2_iso = h_m2_iso/M_m; %[KJ/Kg]
s_m_m2_iso = s_m2_iso/M_m; %[KJ/Kg/K]

%% Thermodynamics @ Compressor Outlet - Real
h_m2_eff=h_m1+((h_m2_iso-h_m1)/eff);
test = 1;

T2= T2_iso;

n=int8(0);
clear diff;
while (test==1)
```

```
    [v_m2, h_m2, s_m2]=Thermodynamic_Properties (T2, P2, x_a2, x_w2);
    diff = h_m2-h_m2_eff;

    if (isreal(diff)==0)
          T2=T2-10;

    elseif (diff<-0.1)
           T2=T2+30;

   elseif ((abs(diff)>200))
           T2=T2-2;
    elseif ((abs(diff)>50))
           T2=T2-1;

    elseif ((abs(diff)>10))
           T2=T2-0.1;

    elseif (abs(diff)>1)
           T2=T2-0.01;

    elseif (abs(diff)>0.1)
           T2=T2-0.0005;


    else
           test=0;

    end


    n=n+1;
        if (n==3000)
            test=0;
            F_S=0;
            msgbox('Did not converge','Compressor - Error');
        end

end

v_m_m2 = v_m2/M_m; %[m^3/kg]
h_m_m2 = h_m2/M_m; %[KJ/Kg]
s_m_m2 = s_m2/M_m; %[KJ/Kg/K]

Work_dot = (h_m_m2 - h_m_m1).*m_dot2;%[KJ/sec]
end
```

```
%% SPRAY #1

function [P3,T3,h_m_m3, s_m_m3, v_m_m3,m_dot3,...
    m_dot_w3, m_dot_a3, x_a3, x_w3, M_m3,F_S] = ...
    Spray1(T2,P2, m_dot_water,m_dot2,T_w,x_a2,x_w2,m_dot_a2,m_dot_w2,F_S)
%[K,Pa, Humidity,Kg/sec]

clear v_m3 h_m3 s_m3 h_m2

% water properties

% T_w = 93.3333C
% P_w = 49 PSIG or 337.8431073652497 kPa

%@ 200kPa and 20 C      @ 500kPa and 20 C              @1MPa and 20C
%v_l=0.001002 m^3/kg    v_l=0.001002 m^3/kg            v_l=0.001001 m^3/kg
%u_l=83.91 KJ/Kg        u_l=83.89 KJ/Kg                u_l=83.86 KJ/Kg
%h_l=84.11 KJ/Kg        h_l=84.39 KJ/Kg                h_l=84.86 KJ/Kg
%s_l=0.2965 KJ/Kg/K     s_l=0.2964 KJ/Kg/K             s_l=0.2963 KJ/Kg/K
%from 2006 Knavel steam table

% The thermodynamic properties of liquid water driven by temperature change
% and almost constant with pressure changes due to the incompressibility
%of the fluid.

% extrapulated values are as follow
%@ 337.843kPa and 20C
%v_l=0.001002 m^3/kg
%u_l=83.90 KJ/Kg
%h_l=84.24 KJ/Kg
%s_l=0.2965 KJ/Kg/K

%Therefore, Saturation values of Liquid water will be used via XSteam
%function.

if (F_S==1)% F_S - Fail safe controller
%% Definitions

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

M_m2=x_a2.*M_a+x_w2.*M_w;
MR = M_w/M_a; %Molar Ratio
%% Injectered Water Properies
T_w =T_w -273.15;
h_w =XSteam('hL_T',T_w); %[KJ/Kg]
H_dot_water = h_w.*m_dot_water;%[KJ/s]

%% After Injector - Mixure Properties
m_dot3 = m_dot_water + m_dot2; %[Kg/sec] - Conservation of mass.
m_dot_w3 = m_dot_water+m_dot_w2;%[Kg/sec]
m_dot_a3 = m_dot_a2;%[Kg/sec]

MF = m_dot_w3/m_dot_a3;
```

```
x_w3=(MF/MR)/((MF/MR)+1);
x_a3=1/((MF/MR)+1);

M_m3=x_a3.*M_a+x_w3.*M_w;% [Kg/Kmol]

[~, h_m2, ~]=Thermodynamic_Properties (T2, P2, x_a2, x_w2);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]
H_dot2 = (h_m2/M_m2).*m_dot2; %[KJ/sec]
H_dot3 = H_dot2 + H_dot_water;%[KJ/sec]
h_m3_eff = (H_dot3/m_dot3).*M_m3; %[KJ/Kmol]


test = 1;
T3=T2;
P3=P2;
n=0;
while (test==1)

    [~, h_m3, ~]=Thermodynamic_Properties (T3, P3, x_a3, x_w3);
    diff = h_m3-h_m3_eff;

    if (isreal(h_m3)==0)
          T3=T3-7;

    elseif (diff<-0.1)
            T3=T3+25;
    elseif ((abs(diff)>1000))
            T3=T3-10;

    elseif ((abs(diff)>500))
            T3=T3-2;

    elseif ((abs(diff)>50))
            T3=T3-0.1;

    elseif (abs(diff)>10)
             T3=T3-0.01;

    elseif (abs(diff)>0.5)
            T3 = T3-0.005;


    else
            test=0;

    end

    n=n+1;
    if (n ==3000)
        test=0;
        F_S=0;
        msgbox('Did not converge','Spray1 - Error');
    end
```

```
end
[v_m3, h_m3, s_m3]=Thermodynamic_Properties (T3, P3, x_a3, x_w3);

s_m_m3= s_m3/M_m3;  %[KJ/Kg/K]
v_m_m3 = v_m3/M_m3; %[M^3/Kg]
h_m_m3 = h_m3/M_m3; %[KJ/Kg]

if((h_m_m3.*m_dot3-H_dot3)/H_dot3>0.001)
    F_S=0;
end
end
```

HEAT EXCHANGER

```
function [T4,P4,T7,P7,Q,F_S] = ...
    Heat_exchanger(P3,T3,P6,T6, m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2...
    ,m_dot_CA_w,m_dot_CA_a,F_S)
%[Pa],[K],[kg/s]

% _Tubeside or CA - 'Clean Air'_
% _Shellside or FG - 'Flue Gas'_

if (F_S==1)% F_S - Fail safe controller
%% Heat Exhcanger Parameters
OD = 0.5; % [in]<-tube deameter
t_tube = 0.035; %[in] <-tube thickness
CC_horiz = 0.75; %[in]<-center to center horizontal
CC_vert = 0.875; %[in]<-center to center vertical
baffle_spacing = 12.5625; %[in]
Ch_width =8.75; %[in]
Ch_hight =7.25; %[in]
num_tube_horiz = 10; % Heater Cross Section
num_tube_vert = 10;
%% Avegare Conditions
r_d = 0.001;

%% Scoped Design
N_1 = 1; % pass
L = 15; %[ft/pass]
N_o = 14; % x passes

A_ht = num_tube_horiz.*num_tube_vert.*N_1.*L.*(pi().*OD)/12;


%% Constant Fluids Properties
Pr_num_CA = 0.74;
Pr_num_FG = 0.74; %Pr=cu/k
atm_density = 0.073; %[lb/ft^3]


%% Conversions
m_dot_CA_im = (m_dot_CA_w+m_dot_CA_a).*(1/0.45359237).*3600; %[lb/hr]
m_dot_FG_im = (m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2).*(1/...
    0.45359237).*3600; %[lb/hr]

ratio_CA = m_dot_CA_w/m_dot_CA_a; %[lb_w/lb_da]

T3_im = (T3-273.15).*9/5+32; %[F]
P3_im = (P3-101325).* 0.000145037738; %[PSIG]

T6_im = (T6-273.15).*9/5+32; %[F]
P6_im = (P6-101325).* 0.000145037738; %[PSIG]

%% CA - INLET
```

```
%%% Percent Ratio of CA Mass.

percent_H2O_CA = ratio_CA/(1+ratio_CA);
% _Assume air consist of 24.5% O_2(Oxigen) and 75.5% N_2(Nitrogen) by
% weight._
percent_O2_CA = (1-percent_H2O_CA).*0.245;
percent_N2_CA = (1-percent_H2O_CA).*0.755;
percent_CO2_CA = 0;
%%% CA Mass Inlet parameters

%Determine the Hydrolic Deameter of the FG(flow gas)
%and CA (Clean Air) flow.

D_hydrolic_CA = (4.*(pi().*(OD-2.*t_tube).^2)/4)/(pi().*(OD-2.*t_tube));
Area_CA = num_tube_horiz.*num_tube_vert.*((pi().*(OD-2.*t_tube).^2)...
    /4)/144; %[ft^2]
% _Area_CA is the surface area that clean air is in contact with._
G_CA = m_dot_CA_im/Area_CA; %[lb/ft^2 hr]


[~,~, ~, Rho_CA_in,Vel_CA_in, ~,~, ~, h_i_CA_in,Delta_P_CA_in] = ...
    Flow_charectoristics (atm_density,T3_im ,P3_im,m_dot_CA_im,Area_CA,...
    G_CA, D_hydrolic_CA, Pr_num_CA, percent_N2_CA,percent_O2_CA,...
    percent_H2O_CA, percent_CO2_CA,Ch_hight);

%% CA - Mass Flow Rate Breakdown
m_dot_CA_H2O = percent_H2O_CA.*m_dot_CA_im;
m_dot_CA_O2 = percent_O2_CA.*m_dot_CA_im;
m_dot_CA_N2 = percent_N2_CA.*m_dot_CA_im;
m_dot_CA_CO2 =percent_CO2_CA.*m_dot_CA_im;

%% FG - INLET
D_hydrolic_FG = 4.*(CC_horiz.*CC_vert -(pi().*OD.^2/4))/(pi().*OD);
Area_FG = (Ch_width-(num_tube_horiz.*OD)).*baffle_spacing/144;

G_FG = m_dot_FG_im/Area_FG;
percent_H2O_FG = (m_dot6_H2O)/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
% chemical composition should be extracted from burner.
percent_O2_FG = m_dot6_O2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
percent_N2_FG = m_dot6_N2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
percent_CO2_FG =m_dot6_CO2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);

[~,~, ~, ~,~, ~,~, ~, h_i_FG_in,Delta_P_FG_in] = Flow_charectoristics ...
    (atm_density,T6_im ,P6_im,m_dot_FG_im,Area_FG, G_FG, D_hydrolic_FG,...
    Pr_num_FG, percent_N2_FG,percent_O2_FG,percent_H2O_FG,...
    percent_CO2_FG,Ch_hight);

%% FG - Mass Flow Rate Breakdown
m_dot_FG_H2O = percent_H2O_FG.*m_dot_FG_im;
m_dot_FG_O2 = percent_O2_FG.*m_dot_FG_im;
m_dot_FG_N2 = percent_N2_FG.*m_dot_FG_im;
m_dot_FG_CO2 =percent_CO2_FG.*m_dot_FG_im;
```

```
test=int8(0);
n=int8(0);

T4_im=T3_im;
T7_im=T6_im;
P4_im = P3_im;
P7_im = P6_im;

while (test==0)
    %% CA
    Q_ave_H2O_CA = (19.86/18).*(T4_im-T3_im)-2.*(597/18).*((T4_im+459.7).^...
        0.5-(T3_im+459.7).^0.5)+(7500/18).*(log(T4_im+459.7)-
log(T3_im+459.7));
    Q_ave_N2_CA = (9.47/24).*(T4_im-T3_im) - (3470/24).*(log(T4_im+459.7)...
        -log(T3_im+459.7))+ (1160000/24).*(1/(T4_im+459.7)-1/(T3_im+459.7));
    Q_ave_O2_CA =  (11.515/32).*(T4_im-T3_im)-
2.*(172/32).*((T4_im+459.7).^...
        0.5-(T3_im+459.7).^0.5)+(1530/32).*(log(T4_im+459.7)-
log(T3_im+459.7));
    Q_ave_CO2_CA = (16.2.*(T4_im-T3_im)-6530.*(log(T4_im+459.7)-log(T3_im...
        +459.7))-1410000.*(1/(T4_im+459.7)-1/(T3_im+459.7)))/44;

    Q_ave_CA = [Q_ave_H2O_CA,Q_ave_N2_CA,Q_ave_O2_CA,Q_ave_CO2_CA]*...
        [m_dot_CA_H2O,m_dot_CA_N2,m_dot_CA_O2,m_dot_CA_CO2]';

    %% FG
    Q_ave_H2O_FG = (19.86/18).*(T6_im-T7_im)-2.*(597/18).*((T6_im+459.7).^...
        0.5-(T7_im+459.7).^0.5)+(7500/18).*(log(T6_im+459.7)-
log(T7_im+459.7));
    Q_ave_N2_FG = (9.47/24).*(T6_im-T7_im) - (3470/24).*(log(T6_im+459.7)-...
        log(T7_im+459.7))+ (1160000/24).*(1/(T6_im+459.7)-1/(T7_im+459.7));
    Q_ave_O2_FG =  (11.515/32).*(T6_im-T7_im)-
2.*(172/32).*((T6_im+459.7).^...
        0.5-(T7_im+459.7).^0.5)+(1530/32).*(log(T6_im+459.7)-
log(T7_im+459.7));
    Q_ave_CO2_FG = (16.2.*(T6_im-T7_im)-6530.*(log(T6_im+459.7)-log(T7_im+...
        459.7))-1410000.*(1/(T6_im+459.7)-1/(T7_im+459.7)))/44;

    Q_ave_FG = [Q_ave_H2O_FG,Q_ave_N2_FG,Q_ave_O2_FG,Q_ave_CO2_FG]*...
        [m_dot_FG_H2O,m_dot_FG_N2,m_dot_FG_O2,m_dot_FG_CO2]';

    %% Average Conditions
    T_ave = (T4_im+T3_im+T7_im+T6_im)/4;
    k_steel = -0.000000495314.*T_ave.^2+0.00545322.*T_ave+8.22022;

    r_w = 1/(k_steel/(t_tube/12));


    [~,~, ~, ~, ~, ~,~, ~, h_i_CA_out,~] = Flow_charectoristics
(atm_density,...
        T4_im ,P4_im,m_dot_CA_im,Area_CA, G_CA, D_hydrolic_CA, Pr_num_CA, ...
        percent_N2_CA,percent_O2_CA,percent_H2O_CA, percent_CO2_CA,Ch_hight);
```

```
    [~,~, ~, ~, ~, ~,~, ~, h_i_FG_out,~] = Flow_charectoristics (atm_density,
...
        T7_im ,P7_im,m_dot_FG_im,Area_FG, G_FG, D_hydrolic_FG, Pr_num_FG,...
        percent_N2_FG,percent_O2_FG,percent_H2O_FG, percent_CO2_FG,Ch_hight);


    U = 1/(2/(h_i_CA_in+h_i_CA_out)+2/(h_i_FG_in+h_i_FG_out)+r_d +r_w);
    if ((T6_im-T4_im)==(T7_im-T3_im))
        LMTD = (T6_im-T4_im);
    else
        LMTD = ((T6_im-T4_im)-(T7_im-T3_im))/log(((T6_im-T4_im)/...
            (T7_im-T3_im)));
    end
    Q_im = U.*A_ht.*LMTD;
    if ((abs(Q_ave_FG-Q_ave_CA)<0.5)&&(abs(Q_im-Q_ave_CA)<0.5)...
            &&(abs(Q_im-Q_ave_FG)<0.5))
        test=1;
    else
        T4_im = T4_im+(Q_im-Q_ave_CA)/1000000;
        T7_im = T7_im-(Q_im-Q_ave_FG)/1000000;

    %% Pressure Drop
    [~,~, ~, Rho_CA_out,Vel_CA_out, ~,~, ~, ~,Delta_P_CA_out] = ...
        Flow_charectoristics (atm_density,T4_im
,P4_im,m_dot_CA_im,Area_CA,...
        G_CA, D_hydrolic_CA, Pr_num_CA, percent_N2_CA,percent_O2_CA,...
        percent_H2O_CA, percent_CO2_CA,Ch_hight);

    Delta_P_CA = (Delta_P_CA_out+Delta_P_CA_in)/2.*(N_1.*L)+4.*(N_1 -1)*...
        (Rho_CA_in.*Vel_CA_in.^2+Rho_CA_out.*Vel_CA_out.^2)/2/32.2/144;

    P4_im = P3_im - Delta_P_CA;

    [~,~, ~, ~,~, ~,~, ~, ~,Delta_P_FG_out] = Flow_charectoristics ...
        (atm_density,T7_im ,P7_im,m_dot_FG_im,Area_FG, G_FG,
D_hydrolic_FG,...
        Pr_num_FG,percent_N2_FG,percent_O2_FG,percent_H2O_FG,...
        percent_CO2_FG,Ch_hight);

    Delta_P_FG = (Delta_P_FG_out+Delta_P_FG_in)/2.*N_o;

    P7_im = P6_im - Delta_P_FG;

    end
    n=n+1;

end


%% Converison
T4 = (T4_im-32).*5/9+273.15; %[K]
P4 = (P4_im+14.696)/0.000145037738; %[Pa]

T7 = (T7_im-32).*5/9+273.15; %[K]
```

```
P7 = (P7_im+14.696)/0.000145037738; %[Pa]
Q = Q_im/3412.14;%[KW]

if ((isreal(T4)==0)||(isreal(T7)==0)||isreal(P4)==0||(isreal(P7)==0))
    T4 = T3+200;
    T7 = T6-200;
    P4=P3;
    P7=P6;
end
else
    F_S =0;
end
%% Flow Charectaristics
function [Visc, Cp, k, density, velocity, Reynolds_num, Colburn_factor, ...
    Friction_factor, h, delta_P] = Flow_charectoristics (atm_density, ...
    T_im ,P_im,m_dot_im,Area, G, D_hydrolic, Pr_num, percent_N2, ...
    percent_O2,percent_H2O, percent_CO2,Ch_hight)

Visc = 2.42.*exp(0.075484.*log(T_im).^2-0.578461.*log(T_im)-2.931705);
%[lb/ft-hr]

Cp= percent_N2.*(9.47-3470/(T_im+459.7)+1160000/(T_im+459.7).^2)/24 ...
    + percent_O2.*(11.515-172/(T_im+459.7).^0.5+1530/(T_im+459.7))/32 ...
    + percent_H2O.*(19.86-597/(T_im+459.7).^0.5+7500/(T_im+459.7))/18 ...
+ percent_CO2.*(11.515-172/(T_im+459.7).^0.5+1530/(T_im+459.7))/32;
%[Btu/lb/F]

k = Cp.*Visc/Pr_num; %[Btu/hr-ft*F]

density = atm_density.*((P_im + 14.696)/14.696)/((T_im+459.63)/519.63);
%[lb/ft^3]
velocity = m_dot_im/density/Area/3600; %[ft/sec]
Reynolds_num = (D_hydrolic/12).* G/Visc;

if (percent_CO2==0)
    if (Reynolds_num<2000)
       Colburn_factor = exp(0.3543232.*log(Reynolds_num)-1.0953035);% j_h
    else
       Colburn_factor = exp(0.0219973.*log(Reynolds_num).^3-...
           0.74268.*log(Reynolds_num).^2+9.0828322.*log(Reynolds_num)...
           -34.1798896);% j_h
    end
else
    Colburn_factor = exp(0.5358039.*log(Reynolds_num)-0.9033314);% j_h
end
if (percent_CO2==0)
   if (Reynolds_num<1000)
        Friction_factor= exp(-0.9964398.*log(Reynolds_num)+...
            8.4994719)/10000;% f[ft^2/in^2]
    else
        Friction_factor= exp(-0.2601568.*log(Reynolds_num)+...
            3.4073562)/10000;% f[ft^2/in^2]
    end
else
    Friction_factor =exp(-0.1876798.*log(Reynolds_num) + 4.7805861 )/10000;
end
```

```
h = Colburn_factor.*k/(D_hydrolic/12).*Pr_num.^(1/3);
% h_i=(j_h*k/D)*Pr^1/3 [Btu/hr F ft^2]

if (percent_CO2==0)
    delta_P = Friction_factor.*G^2/(2.*(32.2.*3600^2).*density.*...
        D_hydrolic/12);
else
    delta_P = Friction_factor.*G^2.*Ch_hight/12/(2.*(32.2.*3600^2).*...
        density.*D_hydrolic/12);
end
```

TURBINE

```
function [h_m_m4, s_m_m4, v_m_m4,P5_iso,T5_iso,h_m_m5_iso, ...
    s_m_m5_iso, v_m_m5_iso,P5,T5,h_m_m5, s_m_m5, v_m_m5,Work_dot,F_S] = ...
    Turbine(T4,P4,m_dot_a,m_dot_w, P_amb,F_S)%[K,Pa,Kg/sec]

%clear v_m4 h_m4 s_m4 v_m5_iso h_m5_iso s_m5_iso v_m5 h_m5 s_m5;

if (F_S==1)% F_S - Fail safe controller
%% Definitions
eff = 0.8; % Turbine thermal efficency

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

MR = M_w/M_a; % Molar Ratio

m_dot = m_dot_w + m_dot_a; %[Kg/sec] - Conservation of mass.

MF = m_dot_w/m_dot_a;

x_w=(MF/MR)/((MF/MR)+1);
x_a=1/((MF/MR)+1);

M_m=x_a.*M_a+x_w.*M_w;%[Kg/Kmol]

%% Thermodynamics @ Turbine Inlet

[v_m4, h_m4, s_m4]=Thermodynamic_Properties (T4, P4, x_a, x_w);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]

v_m_m4 = v_m4/M_m; %[m^3/Kg]
h_m_m4 = h_m4/M_m; %[KJ/Kg]
s_m_m4 = s_m4/M_m; %[KJ/Kg/K]

P5 = ((P_amb.*0.000145037738)+0.5)/0.000145037738;

%% Thermodynamics @ Turbine Outlet -Isothermal

test = 1;

T5_iso = T4;
n=int8(0);
while (test==1)
   n=n+1;
    [v_m5_iso, h_m5_iso, s_m5_iso]=Thermodynamic_Properties (T5_iso, ...
        P5, x_a, x_w);
    diff = s_m5_iso-s_m4;

    if (isreal(diff)==0)
            T5_iso=T5_iso-30;

    elseif (diff<-0.05)
```

```
            T5_iso=T5_iso+50;

    elseif ((abs(diff)>50))
            T5_iso=T5_iso-30;

    elseif ((abs(diff)>10))
            T5_iso=T5_iso-10;

    elseif (abs(diff)>5)
            T5_iso=T5_iso-3;

    elseif (abs(diff)>0.5)
            T5_iso=T5_iso-0.5;

    elseif (abs(diff)>0.05)
            T5_iso=T5_iso- 0.05;

    else
            test =0;

    end


    n=n+1;
        if (n ==3000)
            test =0;
            F_S=0;
            msgbox('Did not converge','Turbine - Error');
        end


end


P5_iso = P5;%[Pa]
v_m_m5_iso = v_m5_iso/M_m; %[m^3/kg]
h_m_m5_iso = h_m5_iso/M_m; %[KJ/Kg]
s_m_m5_iso = s_m5_iso/M_m; %[KJ/Kg/K]

%% Thermodynamics @ Turbine Outlet - Real
h_m5_eff=h_m4-(eff.*(h_m4-h_m5_iso));

test = 1;
n=int8(0);
T5= T5_iso;


while (test==1)

    [v_m5, h_m5, s_m5]=Thermodynamic_Properties (T5, P5, x_a, x_w);
    diff = h_m5_eff-h_m5;

    if (isreal(diff)==0)
```

```
            T5=T5+10;

    elseif (diff<0)
            T5=T5-50;

    elseif ((abs(diff)>500))
            T5=T5+2;

    elseif ((abs(diff)>100))
            T5=T5+1;

    elseif ((abs(diff)>10))
            T5=T5+0.1;

    elseif (abs(diff)>5)
            T5=T5+ 0.01;

    elseif (abs(diff)>0.1)
            T5=T5+ 0.0005;

    else
            test =0;

    end


    n=n+1;
        if (n ==3000)
            test =0;
            F_S=0;
            msgbox('Did not converge','Turbine - Error');
        end

end


v_m_m5 = v_m5/M_m; %[m^3/kg]
h_m_m5 = h_m5/M_m; %[KJ/Kg]
s_m_m5 = s_m5/M_m; %[KJ/Kg/K]

Work_dot = (h_m_m5 - h_m_m4).*m_dot;%[KJ/sec]

end
```

COMBUSTER

```
function
[T6,P6,m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2,Q_total,Q_created,F_S] = ...
    Combustor(T_amb,P5,T5,m_dot5_w,m_dot5_a,T_Fuel,m_dot_Fuel,F_S)
% maximum allowable temeprature.
%T_max = 1350; %[F]

if (F_S==1)% F_S - Fail safe controller


%% Conversion
T5_im = (T5-273.15).*9/5+32; %[F]
%P5_im = (P5-101325).* 0.000145037738; %[PSIG]
T6_im = T5_im;
T_Fuel_im = (T_Fuel-273.15).*9/5+32; %[F]

T_amb_im = (T_amb-273.15).*9/5+32; %[F]


%% Turbine Exit Flow


ratio_TE = m_dot5_w/m_dot5_a;
%%% Percent Ratio of TE Mass.

percent_H2O_TE = ratio_TE/(1+ratio_TE);
% _Assume air consist of 24.5% O_2(Oxigen) and 75.5% N_2(Nitrogen) by
% weight._
percent_O2_TE = (1-percent_H2O_TE).*0.245;
percent_N2_TE = (1-percent_H2O_TE).*0.755;
percent_CO2_TE = 0;
%%% TE Mass parameters
m_dot5_H2O_im = m_dot5_w.*(1/0.45359237).*3600; %[lb/hr]
m_dot5_N2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*percent_N2_TE;
m_dot5_O2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*percent_O2_TE;
m_dot5_CO2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*...
    percent_CO2_TE;
[Q_ave_H2O_TE,Q_ave_N2_TE,Q_ave_O2_TE,Q_ave_CO2_TE] = Q_Average(...
    T_amb_im,T5_im);

Q_ave_TE = [Q_ave_H2O_TE,Q_ave_N2_TE,Q_ave_O2_TE,Q_ave_CO2_TE]*...
    [m_dot5_H2O_im,m_dot5_N2_im,m_dot5_O2_im,m_dot5_CO2_im]';

%% Fuel Flow

m_dot_Fuel_im = m_dot_Fuel.*(1/0.45359237).*3600; %[lb/hr]
% Combuster efficency is estimated to be 100% due to the garantee 100%
% combustion. Heat losses of the system and combuster will be
% calculated independantly.

q_fuel = 21702; %[BTU/lb] of Propane HHV
```

```
Q_fuel_im = q_fuel.*m_dot_Fuel_im; %[BTU/hr]

test = 1;
 n=int8(0);
while (test == 1)
    if (n==3000)
        test =0;
        msgbox('Did not converge','Combuster - Error');
        F_S =0;
    else
        n=n+1;
    end
    %% Product Charactoristics

    m_dot6_CO2_im = (m_dot_Fuel_im/16.042).*(12.0107+2.*15.9994);%[lb/hr];
    m_dot6_N2_im = m_dot5_N2_im;
    m_dot6_H2O_im = (m_dot_Fuel_im/16.042).*(4.*1.00794+2.*15.9994)+ ...
        m_dot5_H2O_im;%[lb/hr]
    m_dot6_O2_im = m_dot5_O2_im - (m_dot_Fuel_im/16.042).*4.*15.9994;


[Q_ave_H2O_FG_thro,Q_ave_N2_FG_thro,Q_ave_O2_FG_thro,Q_ave_CO2_FG_thro]...
        =Q_Average (T5_im, T6_im);
    Q_total_thro = [Q_ave_H2O_FG_thro,Q_ave_N2_FG_thro,Q_ave_O2_FG_thro,...
        Q_ave_CO2_FG_thro]*[m_dot5_H2O_im,m_dot5_N2_im,m_dot5_O2_im,...
        m_dot5_CO2_im]';



    %% Created Temperature

    T_CO2_im = (T_Fuel_im.*12.0107 + T5_im.*2.*15.9994)/44.010;
    T_N2_im = T5_im;
    T_O2_im = T5_im;
    T_H2O_im = (T_Fuel_im.*2.*1.00794 + T5_im*15.9994)/(15.9994+2.*1.00794);

    Q_ave_H2O_FG_created = (19.86/18).*(T6_im-T_H2O_im)-2.*(597/18).*...
        (((T6_im+459.7).^0.5)-((T_H2O_im+459.7).^0.5))+(7500/18).*...
        (log(T6_im+459.7)-log(T_H2O_im+459.7))+(0.4299.*XSteam('hV_p',...
        P5/100000)-0.4299.*XSteam('hL_p',P5/100000));% [BTU/lb]
    Q_ave_N2_FG_created = (9.47/24).*(T6_im-T_N2_im) -
(3470/24).*(log(T6_im...
        +459.7)-log(T_N2_im+459.7))+ (1160000/24).*(1/(T6_im+459.7)-1/...
        (T_N2_im+459.7));
    Q_ave_O2_FG_created =  (11.515/32).*(T6_im-T_O2_im)-
2.*(172/32).*(((T6_im...
        +459.7).^0.5)-((T_O2_im+459.7).^0.5))+(1530/32).*(log(T6_im+459.7)...
        -log(T_O2_im+459.7));
    Q_ave_CO2_FG_created = (16.2.*(T6_im-T_CO2_im)-6530.*(log(T6_im+459.7)...
        -log(T_CO2_im+459.7))-1410000.*(1/(T6_im+459.7)-1/...
        (T_CO2_im+459.7)))/44;

    m_dot_CO2_im_cr = (m_dot_Fuel_im/16.042).*(12.0107+2.*15.9994);%[lb/hr];
    m_dot_N2_im_cr = 0;
```

```
    m_dot_H2O_im_cr =
(m_dot_Fuel_im/16.042).*(4.*1.00794+2.*15.9994);%[lb/hr]
    m_dot_O2_im_cr = -(m_dot_Fuel_im/16.042).*4.*15.9994;
    Q_total_created = [Q_ave_H2O_FG_created,Q_ave_N2_FG_created,...
        Q_ave_O2_FG_created,Q_ave_CO2_FG_created]*[m_dot_H2O_im_cr,...
        m_dot_N2_im_cr,m_dot_O2_im_cr,m_dot_CO2_im_cr]';

    Q_total = (Q_total_thro + Q_total_created);
    diff = (Q_fuel_im - Q_total);
    if (abs(diff)<0.5)
        test =0;
    elseif (diff>0)
        if (diff>50000)
            T6_im = T6_im + 20;
        elseif ((diff<50000)&&(diff>5000))
            T6_im = T6_im + 10;
        elseif ((diff<5000)&&(diff>500))
            T6_im = T6_im + 0.5;
        elseif ((diff<500)&&(diff>50))
            T6_im = T6_im + 0.01;
        else
            T6_im = T6_im + 0.001;
        end
    else
        if (diff<-50000)
            T6_im = T6_im - 20;
        elseif ((diff>-50000)&&(diff<-5000))
            T6_im = T6_im - 10;
        elseif ((diff>-5000)&&(diff<-500))
            T6_im = T6_im - 0.5;
        elseif ((diff>-500)&&(diff<-50))
            T6_im = T6_im - 0.01;
        else
            T6_im = T6_im - 0.001;
        end
    end
end
P6 = P5;
    m_dot6_CO2 = m_dot6_CO2_im.*0.45359237/3600;
    m_dot6_N2= m_dot6_N2_im.*0.45359237/3600;
    m_dot6_H2O = m_dot6_H2O_im.*0.45359237/3600;
    m_dot6_O2 = m_dot6_O2_im.*0.45359237/3600;
    T6 =(T6_im-32).*5/9+273.15; %[K]

 Q_total = (Q_total_thro + Q_total_created)/3412.14;%[KW]
 Q_created = (Q_total_created/3412.14); %[KW,KJ/sec]
end
function [Q_ave_H2O,Q_ave_N2,Q_ave_O2,Q_ave_CO2]=Q_Average (T1, T2)

Q_ave_H2O = (19.86/18).*(T2-T1)-2.*(597/18).*((T2+459.7).^(1/2)-(T1+...
    459.7).^(1/2))+(7500/18).*(log(T2+459.7)-log(T1+459.7));% [BTU/lb]
Q_ave_N2 = (9.47/24).*(T2-T1) - (3470/24).*(log(T2+459.7)-log(T1+...
    459.7))+ (1160000/24).*(1/(T2+459.7)-1/(T1+459.7));
Q_ave_O2 =  (11.515/32).*(T2-T1)-2.*(172/32).*((T2+459.7).^(1/2)-(T1+...
    459.7).^(1/2))+(1530/32).*(log(T2+459.7)-log(T1+459.7));
Q_ave_CO2 = (16.2.*(T2-T1)-6530.*(log(T2+459.7)-log(T1+459.7))-1410000.*...
```

```
(1/(T2+459.7)-1/(T1+459.7)))/44;
```

Thermodynamic Properties

```
function [v_m, h_m, s_m]=Thermodynamic_Properties (T, P, x_a, x_w)

clear B_m dB_m C_m dC_m xx;
% we are solving the following equation. P*v_m/(R*T)=1+B_m/v_m+C_m/v_m^2
% once rearenged: v_m^3-(R*T/P)*v_m^2-(R*T/P)*B_m*v_m-(R*T/P)*C_m = 0
%P(Pa), T(K).
R = 8.314472; %[KJ/Kmol*K] or [(kPa*m^3)/(Kmol*K)]
P_o = 101325; %[pa]
[B_m, dB_m]=Bravo(T,x_a ,x_w);

%B_m [cm^3/mol]

[C_m, dC_m]=Charlie( T,x_a ,x_w);
%C_m [cm^6/mol^2]

xx = roots([1, (-(R.*1000.*(T/P))), (-(R.*1000.*T/P).*B_m/1000),...
    (-(R.*1000.*T/P).*C_m/1000.^2)]);
% adjustments were made to maintaine unit consistancy.

%v_m [m^3/Kmol]
if isreal(xx(1,1))
    v_m = xx(1,1);
elseif isreal(xx(2,1))
            v_m = xx(2,1);
elseif isreal(xx(3,1))
            v_m = xx(3,1);
else v_m = sprintf('error, no real solutions');

end
[h_abs_w, s_abs_w]=Gibb_Steam(P, T); %h_abs_w [KJ/Kmol], s_abs_w [KJ/Kmol/K]
[h_abs_a, s_abs_a]=Air_Properties(T);%h_abs_w [KJ/Kmol], s_abs_w [KJ/Kmol/K]

h_m = x_a.*h_abs_a + x_w.*h_abs_w +R.*T.*((B_m-T.*dB_m)/(v_m.*1000)...
    + (C_m -0.5.*T.*dC_m)/(v_m.*1000).^2);  %[KJ/Kmol]

s_m = x_a.*s_abs_a + x_w.*s_abs_w + R.*(-log(P/P_o)...
    +x_a.*log((P.*(v_m))/(x_a.*R.*1000.*T))...
    + x_w.*log((P.*(v_m))/(x_w.*R.*1000.*T))-((B_m+T.*dB_m)/(v_m.*1000)...
    + (C_m+T.*dC_m)/(2.*(v_m.*1000).^2)));%[KJ/Kmol/K]
clear xx
end
```

Air Properties

```
unction [h_abs, s_abs]=Air_Properties(T) % T[K]
clear h_abs s_abs;
T_o = 273.15; %K

h_abs_const = [28.921316, 2.5861872.*10.^(-3), -1.9010204.*10.^(-5), ...
    5.1208717.*10.^(-8),-3.2775941.*10.^(-11), 8.0457];
T_h = [(T-T_o); (T.^2-T_o.^2)/2;(T.^3-T_o.^3)/3; (T.^4-T_o.^4)/4; (T.^5-
T_o.^5)/5; 1];

h_abs = h_abs_const*T_h; %[KJ/Kmol]

s_abs_const = [28.921316, 2.5861872.*10.^(-3), -1.9010204.*10.^(-5)/2,
5.1208717.*10.^(-8)/3, -3.2775941.*10.^(-11)/4, 0.02393];
T_s = [log(T/T_o);T-T_o; T.^2-T_o.^2;T.^3-T_o.^3; T.^4-T_o.^4; 1];

s_abs = s_abs_const*T_s;% [KJ/Kmol/K]
```

Bravo Function

```
% Script Occcuracy Confirmed!
function [B_m, dB_m, B_aa, B_aw, B_ww, dB_aa, dB_aw, dB_ww]=Bravo(T,x_a ,x_w)
clear B_m dB_m B_aa B_aw B_ww dB_aa dB_aw dB_ww;
%fuction generates B_aa, B_aw, B_ww, dB_aa/dT,
% dB_aw/dT, dB_ww/dT B_m and dB_m/dT in term of T(K), Relative mass of air
% x_a and Relative mass of water x_m.

% B_aa

T_aa= [1, 1/T, 1/T.^2, 1/T.^3, 1/T.^4, 1/T.^5, 1/T.^6];
%T_aa is the columb matrix to address the variable
%componenet of the B_aa eq.

B_aa_1 = [31.831763, -719.51195, -6538137.0, 1.5929828*10.^9,...
    -2.5588842*10.^11, 2.2300382*10.^13, -8.2793465*10.^14];
%B_aa_1 is a Row matrix to address
%the constant component of the B_aa eq.

B_aa = B_aa_1*T_aa';

% B_aw

T_aw = [1, 1/T, 1/T.^2, 1/T.^4];
%T_aw is the columb
%matrix to address the variable componenet of the B_aw eq.

B_aw_1 = [32.366097, -14113.8, -1244535, -0.2348789*10.^10];
%B_aw_1 is a Row matrix to address the constant component of the B_aw eq.

B_aw= B_aw_1*T_aw';

% B_ww

T_ww = [1, 1/T, 1/T.^2, 1/T.^3, 1/T.^4, 1/T.^5, ...
    1/T.^6, 1/T.^7, 1/T.^8, 1/T.^9, 1/T.^10];
%T_ww is the columb matrix to address the variable componenet of the B_aw eq.

B_ww_1 = [-4965.8164, +1.81918860*10.^7, -2.92013995058*10.^10,
2.7032989*10.^13,...
    -1.6045262*10.^16, 6.3750397*10.^18, -1.7206027*10.^21,
3.1222306*10.^23,...
    -3.6643847*10.^25, 2.5256562*10.^27, -7.929348*10.^28];
%B_ww_1 is a Row matrix to address the constant component of the B_aw eq.

B_ww=B_ww_1*T_ww';

% Derivative Functions

% dB_aa
dT_aa= [-1/T.^2, -2/T.^3, -3/T.^4, -4/T.^5, -5/T.^6, -6/T.^7];
```

```
dB_aa_1 = [-719.51195, -6538137.0, 1.5929828*10.^9,...
    -2.5588842*10.^11, 2.2300382*10.^13, -8.2793465*10.^14];

dB_aa = dB_aa_1*dT_aa';

% dB_aw
dT_aw = [-1/T.^2, -2/T.^3, -4/T.^5];

dB_aw_1 = [-14113.8, -1244535, -0.2348789*10.^10];

dB_aw= dB_aw_1*dT_aw';

% dB_ww

dT_ww = [-1/T.^2, -2/T.^3, -3/T.^4, -4/T.^5, ...
    -5/T.^6, -6/T.^7, -7/T.^8, -8/T.^9, -9/T.^10, -10/T.^11];

dB_ww_1 = [+1.81918860*10.^7, -2.92013995058*10.^10, 2.7032989*10.^13,...
    -1.6045262*10.^16, 6.3750397*10.^18, -1.7206027*10.^21,
3.1222306*10.^23,...
    -3.6643847*10.^25, 2.5256562*10.^27, -7.929348*10.^28];

dB_ww=dB_ww_1*dT_ww';

%B_m (cm^3/mol)
B_m = x_a.^2.*B_aa + 2.*x_a.*x_w.*B_aw + x_w.^2.*B_ww;

%dB_m (cm^3/mol/K)
dB_m = x_a.^2.*dB_aa + 2.*x_a.*x_w.*dB_aw + x_w.^2.*dB_ww;
end
```

Charlie Function

```
%confirmed script accuracy
function [C_m, dC_m, C_www, C_aaa, dC_aaa, dC_www]=Charlie( T,x_a ,x_w)
%fuction generates C_m, dC_m, C_www, C_aaa, dC_aaa, dC_wwwin term of
Temparature [K], molar proportion of air [x_a] and water [x_w] in the
mixutre.
clear C_m dC_m C_www C_aaa dC_aaa dC_www;

%C_aaa
T_aaa = [1, 1/T, 1/T.^2, 1/T.^3, 1/T.^4, 1/T.^5, 1/T.^6]; %T_aaa is the
columb
                                                          %matrix to address
the variable componenet
                                                          %of the C_aaa eq.
C_aaa_1 =[1297.5378, 46021.328, 40813154.0,...
    -3.202391*10.^9, +2.2964785*10.^11, -5.3683467*10.^12, -
2.1183915*10.^14]; %C_aaa_1 is a Row matrix to address

%the constant component of the C_aaa eq.
C_aaa = C_aaa_1*T_aaa';

%C_aaw
T_aaw = [1, 1/T, 1/T.^2, 1/T.^3, 1/T.^4]; %T_aaw is the columb
                                          %matrix to address the variable
componenet
                                          %of the C_aaw eq.
C_aaw_1 = [482.737, 105678, -6.56394*10.^7, 2.94442*10.^10, -3.19317*10.^12];
%C_aaw_1 is a Row matrix to address

%the constant component of the C_aaw eq.
C_aaw = C_aaw_1*T_aaw';

%C_aww
T_aww = [1, 1/T, 1/T.^2, 1/T.^3]; %T_aww is the columb
                                  %matrix to address the variable componenet
                                  %of the C_aww eq.
C_aww_1 = [-10.72887, 3478.04, -383383, 33406000];%C_aww_1 is a Row matrix to
address
                                                  %the constant component of
the C_aww eq.
C_aww = -10.^6*exp(C_aww_1*T_aww');

%C_www
T_www = [1, T, T.^2, T.^3, T.^4, T.^5, T.^6];%T_www is the columb
                                                          %matrix to address
the variable componenet
                                                          %of the C_www eq.
C_www_1 = [-6.566276606, 0.3894679516, -0.0034281020537, 1.333924918*10.^-
5,...
    -2.726404078*10.^-8, 2.839369136*10.^-11, -1.189114330*10.^-14]; %C_www_1
is a Row matrix to address
                                                                  %the

%constant component of the C_www eq.
```

```
C_www = -10.^6* exp(C_www_1*T_www');

% Derivative Functions

%dC_aaa
dT_aaa = [-1/T.^2, -2/T.^3, -3/T.^4, -4/T.^5, -5/T.^6, -6/T.^7];

dC_aaa_1 =[46021.328, 40813154.0,...
    -3.202391*10.^9, +2.2964785*10.^11, -5.3683467*10.^12, -
2.1183915*10.^14];

dC_aaa = dC_aaa_1*dT_aaa';

%dC_aaw

dT_aaw = [ -1/T.^2, -2/T.^3, -3/T.^4, -4/T.^5];

dC_aaw_1 = [105678, -6.56394*10.^7, 2.94442*10.^10, -3.19317*10.^12];

dC_aaw = dC_aaw_1*dT_aaw';

%dC_aww

dT_aww = [ -1/T.^2, -2/T.^3,-3/T.^4];

dC_aww_1 = [3478.04, -383383, 33406000];

dC_aww = C_aww.*(dC_aww_1*dT_aww');

%dC_www

dT_www = [1, 2.*T, 3.*T.^2, 4.*T.^3, 5.*T.^4, 6.*T.^5];

dC_www_1 = [0.3894679516, -0.0034281020537, 1.333924918*10.^-5,...
    -2.726404078*10.^-8, 2.839369136*10.^-11, -1.189114330*10.^-14];

dC_www = C_www*(dC_www_1*dT_www');

%C_m (cm^6/mol^2)
C_m = (x_a.^(3).*C_aaa + (3.*(x_a.^2.*x_w.*C_aaw)) +
(3.*(x_a.*x_w.^(2).*C_aww)) +x_w.^(3).*C_www);

%dC_m (cm^6/mol^2/K)

dC_m = (x_a.^(3).*dC_aaa + (3.*(x_a.^2.*x_w.*dC_aaw)) +
(3.*(x_a.*x_w.^(2).*dC_aww)) +x_w.^(3).*dC_www);
end
```

Gibbs Function

```
function [h_abs, s_abs, Gibbs_Const, Spesific_volume]=Gibb_Steam(P, T) %
P[Pa], T[K]
clear h_abs s_abs Gibbs_Const Spesific_volume;
P_star = 1000000; %[Pa]
P_pi = P/P_star;
T_star = 540; %[K]
Tau= T_star/T;

R = 8.3144; %[KPa m^3 Kmol^-1 K^-1]

n = [-9.69276865,   10.08665597,    -0.005608791,   0.071452738,    -
0.407104982,   1.424081917,    -4.383951132,   -0.284086325,   0.021268464];
J = [0, 1,  -5, -4, -3, -2, -1, 2,  3];
Tau_=Tau*ones(1,9);
Tau1 = Tau_.^J;
Sum1 = Tau1*n';

Gamma = log(P_pi)+Sum1; %In matlab ln(x) is expressed log(x) and log (x)
expressed log10(x)

Gibbs_Const = Gamma.*R.*T;%Unitless

Spesific_volume = R.*T/(P/1000); %[m^3/Kmol]
Tau2 =Tau.*ones(1,9);

sum2 = (J.*Tau2.^(J-1))';
sum_h = n*sum2;
h_abs =R.*T.*Tau.*sum_h; %[KJ/Kmol]

s_abs =R.*(sum_h.*Tau-Gamma);%[KJ/Kmol/K]
end
```

HBC SIMULATION - Humidity Variation Test G-Model

```
%% HBC SIMULATION - Humidity Veration Test
%%
F_S=1;% F_S - Fail safe controller, design to complete the program if non
%convergence is occurs.

%% Inlet Conditions
% Air
Divider_i = 6; % should be 6
T_amb = 288.70556; % [K]
P_amb = 101325; %[Pa]
m_dot = (2432.06.*0.45359237)/3600; % [kg/s]
Phi = 0.5; % percent humidity
X_water_1 = 1;
% water
m_dot_water =(133.5.*0.45359237)/3600;%[kg/s]
T_w = 20+273.15; %[K]

%global P_range_s P_range_l M_size
%M_size = 1000;

%[P_range_s,P_range_l] = Initial_W_Matrix_G(1005, 0.00001, 1800, 275,M_size);
%Fuel

T_Fuel = 290;%[K]
m_dot_fuel = (23.13.*0.45359237/3600);%-0.0017; %[kg/s]
% Arbitraty min fuel used as a low point.

%% Exturnal Loop
Phi_ind = 0.0000001; % Phi indicator


T= zeros(20,7,Divider_i);
P= zeros(20,7,Divider_i);

% T_sat = zeros(20,7);

Sys_eff =zeros(20,Divider_i);
Cyc_eff = zeros(20,Divider_i);
fuel_Sys= zeros(20,Divider_i);
water_Sys= zeros(20,Divider_i);
X_water = zeros(20,5,Divider_i);
h_m_m = zeros(20,8,Divider_i);
percent_mass_conserv = zeros(20,Divider_i);
percent_energy_conserv =zeros(20,Divider_i);
CR_final =zeros(Divider_i,1);

CR_max = 3.5;

for i =6:Divider_i
m= 10;
CR = (CR_max-2)+((2/Divider_i).*i);
CR_final(i,1)=CR;
```

```
    while (m <= 20)

       if (F_S==1)% F_S - Fail safe controller
        %% Loop
        % initial conditions

        P6 = P_amb/1000;
        T6_test = 850;
        m_dot_fuel1=(m_dot_fuel-m_dot_fuel.*(0.05*(double(m)-1)))+0.0017;
        % 0.0017kg/s is an approximate amount of fuel required to have
positive
        % efficency of the system w/ m_dot_water
=(133.5.*0.45359237)/3600;%[kg/s].

        m_dot_water1 = m_dot_water;
        if (m_dot_fuel1==0)
            m_dot_fuel1=0.00001;
        end
        m_dot6_CO2 = 0.01.*0.45359237/3600;
        m_dot6_N2= m_dot.*0.78;
        m_dot6_H2O =m_dot_water1;
        m_dot6_O2 =m_dot.*0.22;

        T_calc= zeros(6,7);
        P_calc= zeros(6,7);
        T6 = T6_test;
        test = 0;
        test2 = 0;
        n =1;

        % argument
        while (test==0)

            %% Compressor
            [P1,T1,h_m_m1, ~, ~,~,~,~, ...
                ~,~,P2,T2,h_m_m2, ~,~,Comp_Work_In,m_dot2,...
                m_dot_w2, m_dot_a2, x_a2, x_w2, M_m, X_water_2,F_S] = ...
                Compressor_G(CR,T_amb,P_amb, Phi,m_dot,X_water_1,F_S);

            %% Spray #1
            [P3,T3,h_m_m3, ~, ~,m_dot3,...
                m_dot_w3, m_dot_a3, x_a3, x_w3, M_m3,X_water_3,F_S] = ...
                Spray1_G1(T2,P2,
m_dot_water1,m_dot2,T_w,x_a2,x_w2,m_dot_a2,m_dot_w2,F_S);

            %% Heat Exhanger (HE)

                  %  P3_im = 47.6; %[PSIG]
                   % P3 = (P3_im+14.696)/0.000145037738; %[Pa]

                   % T3_im = 167; %F
                   % T3 = (T3_im-32).*5/9+273.15; %[K]
```

```
                    %m_dot_CA_w = 0.0196;


                    % m_dot_CA_a = 0.3037;

            % pressure drop due to pipes.
            P3_HE =((P3.*0.145037738)-0.5)/0.145037738;

                   % m_dot6_CO2 = 63.59.*0.45359237/3600;
                   % m_dot6_N2= 1826.05.*0.45359237/3600;
                   % m_dot6_H2O = (146.96+52.03).*0.45359237/3600;
                   % m_dot6_O2 = (592.56-92.50).*0.45359237/3600;

                   % P6_im = 1; %PSIG
                   % P6 = (P6_im+14.696)/0.000145037738; %[Pa]

                   % T6_im = 1350; %F
                   % T6 = (T6_im-32).*5/9+273.15; %[K]


                    [T4,P4,T7,P7,Q_HE,F_S] = ...
                Heat_exchanger_G((P3_HE),T3,(P6),T6,
m_dot6_H2O,m_dot6_CO2,...
                m_dot6_N2,m_dot6_O2,m_dot_w3,m_dot_a3,F_S);


                T_phase_4= Water_Vapor_Saturation_Temp(x_w3.*P4);

                        if ((T4 -T_phase_4)/T4>0.0001)
                               X_water_4 = 1;

                        elseif(((T4 -T_phase_4)/T4)<-0.0001)

                               X_water_4=0;

                        else X_water_4 =inf();
                        end
            %% Turbine

            [h_m_m4, ~, ~,~,~,~,~, ~,P5,T5,h_m_m5, ~,
~,Turbine_Work_Out,X_water_5,F_S] = Turbine_G(T4,P4,...
                m_dot_a3,m_dot_w3,P_amb,X_water_4,F_S);

            %% Combuster


[T6,P6,m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2,Q_thro,Q_created,Q_total,F_S
] = ...

Combustor_G(T_amb,P5,T5,m_dot_w3,m_dot_a3,T_Fuel,m_dot_fuel1,F_S);

            if (T6 >1100)
               T6 = T6.*0.9;
```

```
            m_dot_fuel1= m_dot_fuel1 .*0.9;
        end
    %Combustor_G(P5,T5,m_dot_w3,m_dot_a3,T_Fuel,m_dot_fuel1,F_S);



            if (test2==1)
                test =1;
                P_calc(1,:) = [NaN; NaN; NaN; NaN; NaN; NaN; NaN];
                T_calc(1,:) = [NaN; NaN; NaN; NaN; NaN; NaN; NaN];


            else

                n=n+1

                P_calc(n,:) = [P1; P2; P3_HE; P4; P5; P6; P7];
                T_calc(n,:) = [T1; T2; T3; T4; T5; T6; T7];

                P_diff = (P_calc(n,:)-P_calc((n-1),:))./(P_calc(n,:))
                T_diff = (T_calc(n,:)-T_calc((n-1),:))./(T_calc(n,:))
                chk_P = double((ones(1,7).*0.001));
                chk_T = double((ones(1,7).*0.01));

                % Test for tolerance of 0.1% for Pressure and 1% for
temp.
                if (abs(P_diff) < chk_P)
                    if (abs(T_diff) < chk_T)
                    test2=1;% test2 confirms that tolerance is achieved.
                    end
                end

            end

    end

    % Conservation of mass test


     F_S = ConfirmationofLaws(m_dot2,m_dot3, m_dot_water1, ...
       m_dot_fuel, m_dot6_H2O, m_dot6_CO2, m_dot6_N2, m_dot6_O2); %,...


    if (F_S==0)
        m=30;
    else



        P(m,:,i) = P_calc(n,:);
        T(m,:,i) = T_calc(n,:);

         h_m_m(m,:,i)= [ (h_m_m1.*m_dot); (h_m_m2.*m_dot2); ...
```

```
            (h_m_m3.*m_dot2); (h_m_m4.*m_dot2); (h_m_m5.*m_dot2);...
            Q_thro; Q_created; Q_HE];% [KJ/sec]

            X_water(m,:,i) = [X_water_1; X_water_2;X_water_3; X_water_4;
X_water_5];

            Sys_eff(m,i) = (-(Comp_Work_In.*1.05)-
(0.95.*Turbine_Work_Out))...
                /Q_total;
            Cyc_eff (m,i)= (-(Comp_Work_In.*1.05)-
(0.95.*Turbine_Work_Out))...
                /Q_HE;
            fuel_Sys(m,i) = m_dot_fuel1;
            water_Sys(m,i) = m_dot_water1;



        end
            m=m+1
       end
    end

end
Sys_eff_plt = Sys_eff.*100;
Cyc_eff_plt = Cyc_eff.*100;

figure
for i=1:Divider_i
plot(water_Sys(:,i),Sys_eff_plt,'--ok');
title('Water Injection Variation effects on System`s Efficency');
xlabel('Water Injection Rate [Kg/s]');
ylabel('System Efficency');
end

figure
for i=1:Divider_i
plot(water_Sys(:,i),Cyc_eff_plt,'--ok');
title('Water Injection Variation affects on Cycle`s Efficency')
xlabel('Water Injection Rate [Kg/s]');
ylabel('Cycle Efficency');
end
clear P1 P2 P3 P3_HE P4 P5 P6 P7;
clear T1 T2 T3 T4 T5 T6 T7;
```

Compressor Function

```
% Cycle stage 1-2 w/ Inlet temperature drop.

function [P1,T1,h_m_m1, s_m_m1, v_m_m1,P2_iso,T2_iso,h_m_m2_iso, ...
    s_m_m2_iso, v_m_m2_iso,P2,T2,h_m_m2, s_m_m2, v_m_m2,Work_dot,m_dot2,...
    m_dot_w2, m_dot_a2, x_a2, x_w2, M_m,X_water_2, F_S] = ...
    Compressor_G(CR,T_amb,P_amb, Phi,m_dot,X_water_1,F_S)%[K,Pa,%
Humidity,Kg/sec]

clear v_m1 h_m1 s_m1 v_m2_iso h_m2_iso s_m2_iso v_m2 h_m2 s_m2;
if (F_S==1)% F_S - Fail safe controller
%% Definitions
P1 = ((P_amb.*0.000145037738)-0.5)/0.000145037738; %1 Pa =0.000145037738 Psi
                        %0.5 PSIG inlet pressure drop due to filter.
P1 = P1/1000; % change KPa
T1 = T_amb;
eff = 0.8; % Compressor thermal efficiency

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

%% Initial Conditions
[x_sat_w]=Max_Sat_Suit_G(T_amb,P1);%K, KPa
x_w1= x_sat_w.*Phi;
x_a1= 1-x_w1;


m_dot_a1 = m_dot.*x_a1;%[Kg/sec]
m_dot_w1 = m_dot - m_dot_a1;%[Kg/sec]
m_dot2 = m_dot_w1 + m_dot_a1; %[Kg/sec] - Conservation of mass.

m_dot_a2= m_dot_a1;
m_dot_w2= m_dot_w1;
M_m=x_a1.*M_a+x_w1.*M_w;%[Kg/Kmol]

%% Thermodynamics @ Compressor Inlet
x_a2 = x_a1;
x_w2 = x_w1;
[v_m1, h_m1, s_m1]=Thermodynamic_Properties_G1 (T1, P1, x_a1,
x_w1,X_water_1);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]

v_m_m1 = v_m1/M_m; %[m^3/Kg]
h_m_m1 = h_m1/M_m; %[KJ/Kg]
s_m_m1 = s_m1/M_m; %[KJ/Kg/K]

P2 = P1*CR; % Pressure Ratio
%% Thermodynamics @ Compressor Outlet - Isotropic

T2_iso = T1.*3;
T_phase_2= Water_Vapor_Saturation_Temp(x_w2.*P2*1000);
```

```
        if ((T2_iso -T_phase_2)/T2_iso>0.0001)
               X_water_2 = 1;

        elseif((T_2_iso -T_phase_2)/T2_iso<-0.0001)

               X_water_2=0;

        else X_water_2 =inf();
        end
    [~, ~, s_m2_iso]=Thermodynamic_Properties_G1 (T2_iso, ...
        P2, x_a2, x_w2, X_water_2);
    diff = (s_m2_iso-s_m1)/s_m1;

    n=0;
    T_top = T2_iso+200;
    T_bottom = 273.15;

    while (abs(diff)>0.001)
        n=n+1;
        T_mid = (T_top -T_bottom)/2 + T_bottom;

        if ((T_mid -T_phase_2)/T_mid>0.0001)
               X_water_2 = 1;

        elseif((T_mid -T_phase_2)/T_mid<-0.0001)

               X_water_2=0;

        else X_water_2 =inf();
        end
        [~, ~, s_m_mid]=Thermodynamic_Properties_G1 (T_mid, ...
        P2, x_a2, x_w2,X_water_2);
        diff = (s_m1-s_m_mid)/s_m_mid;
        if(n ==3000)
        diff =0;
        end
        if (abs(diff)<0.001)
            T2_iso = T_mid;
        elseif((s_m_mid-s_m1)<0)
            T_bottom = T_mid;
        else
            T_top = T_mid;
        end
    end
[v_m2_iso, h_m2_iso, s_m2_iso]=Thermodynamic_Properties_G1 (T2_iso, ...
        P2, x_a2, x_w2,X_water_2);



        n=n+1;
        if (n ==3000)
            F_S=0;
            msgbox('Did not converge','Compressor - Error');
```

```
        end



P2_iso = P2;%[KPa]
v_m_m2_iso = v_m2_iso/M_m; %[m^3/kg]
h_m_m2_iso = h_m2_iso/M_m; %[KJ/Kg]
s_m_m2_iso = s_m2_iso/M_m; %[KJ/Kg/K]

%% Thermodynamics @ Compressor Outlet - Real
h_m2_eff=h_m1+((h_m2_iso-h_m1)/eff);
T2= T2_iso;
n=0;

[~, h_m2, ~]=Thermodynamic_Properties_G1 (T2, P2, x_a2, x_w2,X_water_2);
    diff = (h_m2_eff-h_m2)/h_m2_eff;

    T_top = T2+200;
    T_bottom = 273.15;

    while (abs(diff)>0.001)
        n=n+1;
        T_mid = (T_top -T_bottom)/2 + T_bottom;


            if ((T_mid -T_phase_2)/T_mid>0.0001)
                   X_water_2 = 1;

            elseif((T_2_mid -T_phase_2)/T_mid<-0.0001)

                   X_water_2=0;

            else X_water_2 =inf();

            end
        [~, h_m_mid, ~]=Thermodynamic_Properties_G1 (T_mid, ...
        P2, x_a2, x_w2,X_water_2);
        diff = (h_m2_eff-h_m_mid)/h_m2_eff;
        if(n ==3000)
        diff =0;
        end
        if (abs(diff)<0.001)
            T2 = T_mid;
        elseif((h_m_mid-h_m2_eff)<0)
            T_bottom = T_mid;
        else
            T_top = T_mid;
        end
    end
    [v_m2, h_m2, s_m2]=Thermodynamic_Properties_G1 (T2, P2, x_a2, x_w2,
X_water_2);
```

```
        if (n==3000)
            F_S=0;
            msgbox('Did not converge','Compressor - Error');
        end

v_m_m2 = v_m2/M_m; %[m^3/kg]
h_m_m2 = h_m2/M_m; %[KJ/Kg]
s_m_m2 = s_m2/M_m; %[KJ/Kg/K]

Work_dot = (h_m_m2 - h_m_m1).*m_dot2;%[KJ/sec]
end
```

SPRAY Function

```
function [P3,T3,h_m_m3, s_m_m3, v_m_m3,m_dot3,...
    m_dot_w3, m_dot_a3, x_a3, x_w3, M_m3,X_water_3,F_S] = ...
    Spray1_G(T2,P2, m_dot_water,m_dot2,T_w,x_a2,x_w2,m_dot_a2,m_dot_w2,F_S)
%[K,Pa,% Humidity,Kg/sec]

clear v_m3 h_m3 s_m3 h_m2

% water properites

% T_w = 93.3333C
% P_w = 49 PSIG or 337.8431073652497 kPa


if (F_S==1)% F_S - Fail safe controller
%% Definitions

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

M_m2=x_a2.*M_a+x_w2.*M_w;
MR = M_w/M_a; %Molar Ratio
%% Injectered Water Properies
P_w = ((P2.*0.145037738)+60)/0.145037738; % Water is injected at 60 PSIG.

[~,~,~,h_w,~,~]=Property_Iterator_G('water',P_w,T_w);
h_w_tild = h_w/M_w;
H_dot_water = h_w_tild.*m_dot_water;%[KJ/s]

%% After Injector - Mixure Properties
m_dot3 = m_dot_water + m_dot2; %[Kg/sec] - Conservation of mass.
m_dot_w3 = m_dot_water + m_dot_w2;%[Kg/sec]
m_dot_a3 = m_dot_a2;%[Kg/sec]

MF = m_dot_w3/m_dot_a3;
x_w3=(MF/MR)/((MF/MR)+1);
x_a3=1/((MF/MR)+1);

M_m3=x_a3.*M_a+x_w3.*M_w;% [Kg/Kmol]
T_phase_2= Water_Vapor_Saturation_Temp(P2*1000);

        if ((T2 -T_phase_2)/T2>0.0001)
               X_water_mid = 1;

        elseif((T2 -T_phase_2)/T2<-0.0001)

               X_water_mid=0;

        end

[~, h_m2, ~]=Thermodynamic_Properties_G1 (T2, P2, x_a2, x_w2,1);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]
```

```
H_dot2 = (h_m2/M_m2).*m_dot2; %[KJ/sec]
H_dot3 = H_dot2 + H_dot_water;%[KJ/sec]
h_m3_eff = (H_dot3/m_dot3).*M_m3; %[KJ/Kmol]

T3=T2;
P3=P2;

n=0;

    T_top = T3;
    % In order to make this model more flexible, T_top  adjusted
    % based on the % water injected in attept to bring it closer to the
    % actual tempeture.
    T_bottom = 285;

    T_phase= Water_Vapor_Saturation_Temp(P3.*x_w3.*1000);
    X_water_top = 1;
    X_water_bottom = 0;
    % saturation Region
    if(T_phase<285)
        T_phase =285;
    end
        [~, h_m_phase_top, ~]=Thermodynamic_Properties_G1 (T_phase, ...
        P3, x_a3, x_w3, X_water_top);
        [~, h_m_phase_bottom, ~]=Thermodynamic_Properties_G1 (T_phase, ...
        P3, x_a3, x_w3, X_water_bottom);

diff =1;
diff1=1;
T_mid1=0;

        if ((h_m_phase_top>h_m3_eff)&&(h_m_phase_bottom<h_m3_eff))

            X_water_mid =0.5;
            while (abs(diff>0.001))

                [~, h_m_mid, ~]=Thermodynamic_Properties_G1 (T_phase, ...
                  P3, x_a3, x_w3, X_water_mid);

                    diff = abs((h_m3_eff-h_m_mid)/h_m_mid);

                if (diff<0.001)

                    T3 = T_phase;
                    X_water_3 = X_water_mid;

                elseif (h_m3_eff>h_m_mid)
                    X_water_bottom = X_water_mid;
                    X_water_mid = X_water_bottom+(X_water_top-
X_water_bottom)/2;
                elseif (h_m3_eff<h_m_mid)
                    X_water_top= X_water_mid ;
                X_water_mid = X_water_bottom+(X_water_top-X_water_bottom)/2;
                end
```

```
          n=n+1;
             if(n ==3000)
             diff =0;
             end
    end

elseif ((h_m_phase_top<h_m3_eff))
  % range above saturation Temp.
  T_bottom = T_phase;
  T_mid = (T_top -T_bottom)/2 + (T_bottom);

  X_water_mid = 1;


    while (abs(diff)>0.01)
    [~, h_m_mid, ~]=Thermodynamic_Properties_G1 (T_mid, ...
    P3, x_a3, x_w3, X_water_mid);


             diff = (h_m3_eff-h_m_mid)/h_m3_eff;

             if (abs(diff)<0.01)
                 T3 = T_mid;
                 X_water_3 = X_water_mid;

             elseif(h_m_mid<h_m3_eff)
                 T_bottom = T_mid;
                 %h_m_bottom = h_m_mid;
                 T_mid = (T_top -T_bottom)/2 + (T_bottom);
             else
                 T_top =  T_mid;
                 %h_m_top = h_m_mid;
                 T_mid = (T_top -T_bottom)/2 + (T_bottom);
             end
              n=n+1;
             if(n ==3000)
             diff =0;
             end
             if(abs(diff1)>abs(diff))
                 diff1=diff;
                 T_mid1 =T_mid;
             end
             if(abs(T_top-T_bottom) < 0.01)
                  diff =0;
                T3 = T_mid1;
                 X_water_3 = X_water_mid;
             end
    end
elseif (h_m_phase_bottom>h_m3_eff)
  % range above saturation Temp.
  T_top = T_phase;
  T_mid = (T_top -T_bottom)/2 + (T_bottom);

  X_water_mid = 0;
```

```
        while (abs(diff)>0.01)
            [~, h_m_mid, ~]=Thermodynamic_Properties_G1 (T_mid, ...
            P3, x_a3, x_w3, X_water_mid);


                diff = (h_m3_eff-h_m_mid)/h_m3_eff;

                if (abs(diff)<0.01)
                    T3 = T_mid;
                    X_water_3 = X_water_mid;

                elseif(h_m_mid<h_m3_eff)
                    T_bottom = T_mid;

                     T_mid = (T_top -T_bottom)/2 + (T_bottom);
                else
                    T_top =  T_mid;

                    T_mid = (T_top -T_bottom)/2 + (T_bottom);
                end
                if(abs(diff1)>abs(diff))
                    diff1=diff;
                    T_mid1 =T_mid;
                end
                if(abs((T_top-T_bottom)) < 0.01)
                     diff =0;
                   T3 = T_mid1;
                    X_water_3 = X_water_mid;
                end
           n=n+1;
                if(n ==3000)
                diff =0;
                end
        end
    else

         F_S=0;
         msgbox('Did not converge','Spray1 - Error');
         diff=0;

    end
```

```
[v_m3, h_m3, s_m3]=Thermodynamic_Properties_G1 (T3, P3, x_a3, x_w3,
X_water_3);
```

```
s_m_m3= s_m3/M_m3;  %[KJ/Kg/K]
v_m_m3 = v_m3/M_m3; %[M^3/Kg]
h_m_m3 = h_m3/M_m3; %[KJ/Kg]

x = (h_m_m3.*m_dot3-H_dot3)/H_dot3
if((h_m_m3.*m_dot3-H_dot3)/H_dot3>0.01)
    F_S=0;
end
end
```

HEAT EXCHANGER Function

```
function [T4,P4,T7,P7,Q,F_S] = ...
    Heat_exchanger_G(P3,T3,P6,T6,
m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2...
    ,m_dot_CA_w,m_dot_CA_a,F_S)
%[Pa],[K],[kg/s]

% _Tubeside or CA - 'Clean Air'_
% _Shellside or FG - 'Flue Gas'_
% Assume Re =1000;
if (F_S==1)% F_S - Fail safe controller
%% Heat Exchanger Parameters
OD = 0.5; % [in]<-tube diameter
t_tube = 0.035; %[in] <-tube thickness
%CC_vert = 0.75; %[in]<-center to center vertical
%CC_horiz = 0.875; %[in]<-center to center horizontal
CC_vert = 0.875; %[in]<-center to center vertical
CC_horiz = 0.75; %[in]<-center to center horizontal
baffle_spacing = 12.5625; %[in]
Ch_width =8.75; %[in]
Ch_hight =7.25; %[in]
num_tube_horiz = 10; % Heater Cross Section
num_tube_vert = 10;
%% Avegare Conditions
r_d = 0.001;

%% Scoped Design
N_1 = 1; % pass
L = 15; %[ft/pass]
N_o = 14; % # of  passes

A_ht = num_tube_horiz.*num_tube_vert.*N_1.*L.*(pi().*OD)/12;% ft^2


%% Constant Fluids Properties
Pr_num_CA = 0.74;
Pr_num_FG = 0.74; %Pr=cu/k
atm_density = 0.073; %[lb/ft^3]


%% Conversions
m_dot_CA_im = (m_dot_CA_w+m_dot_CA_a).*(1/0.45359237).*3600; %[lb/hr]
m_dot_FG_im = (m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2).*(1/...
    0.45359237).*3600; %[lb/hr]

ratio_CA = m_dot_CA_w/m_dot_CA_a; %[lb_w/lb_da]

T3_im = (T3-273.15).*9/5+32; %[F]
P3_im = (P3-101.325).* 0.145037738; %[PSIG]

T6_im = (T6-273.15).*9/5+32; %[F]
P6_im = (P6-101.325).* 0.145037738; %[PSIG]
```

```
%% CA - INLET


%%% Percent Ratio of CA Mass.

percent_H2O_CA = ratio_CA/(1+ratio_CA);
% _Assume air consist of 24.5% O_2(Oxigen) and 75.5% N_2(Nitrogen) by
% weight._
percent_O2_CA = (1-percent_H2O_CA).*0.245;
percent_N2_CA = (1-percent_H2O_CA).*0.755;
percent_CO2_CA = 0;
%%% CA Mass Inlet parameters

%Determine the Hydrolic Deameter of the FG(flow gas)
%and CA (Clean Air) flow.

D_hydrolic_CA = (4.*(pi().*(OD-2.*t_tube).^2)/4)/(pi().*(OD-2.*t_tube));
Flow_Area_CA = num_tube_horiz.*num_tube_vert.*((pi().*(OD-2.*t_tube).^2)...
    /4)/144; %[ft^2]
% Flow_Area_CA is the surface area that clean air is in contact with._
G_CA = m_dot_CA_im/Flow_Area_CA; %[lb/ft^2 hr]


[~,~, ~, Rho_CA_in,Vel_CA_in, ~,~, ~, h_i_CA_in,Delta_P_CA_in] = ...
    Flow_charectoristics (atm_density,T3_im
,P3_im,m_dot_CA_im,Flow_Area_CA,...
    G_CA, D_hydrolic_CA, Pr_num_CA, percent_N2_CA,percent_O2_CA,...
    percent_H2O_CA, percent_CO2_CA,Ch_hight);

%% CA - Mass Flow Rate Breakdown
m_dot_CA_H2O = percent_H2O_CA.*m_dot_CA_im;%lb/hr
m_dot_CA_O2 = percent_O2_CA.*m_dot_CA_im;
m_dot_CA_N2 = percent_N2_CA.*m_dot_CA_im;
m_dot_CA_CO2 =percent_CO2_CA.*m_dot_CA_im;

%% FG - INLET
D_hydrolic_FG = 4.*(CC_horiz.*CC_vert -(pi().*OD.^2/4))/(pi().*OD);
Flow_Area_FG = (Ch_width-(num_tube_horiz.*OD)).*baffle_spacing/144;

G_FG = m_dot_FG_im/Flow_Area_FG;
percent_H2O_FG = (m_dot6_H2O)/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
percent_O2_FG = m_dot6_O2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
percent_N2_FG = m_dot6_N2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);
percent_CO2_FG =m_dot6_CO2/(m_dot6_H2O+m_dot6_CO2+m_dot6_N2+m_dot6_O2);

[~,~, ~, ~,~, ~,~, ~, h_i_FG_in,Delta_P_FG_in] = Flow_charectoristics ...
    (atm_density,T6_im ,P6_im,m_dot_FG_im,Flow_Area_FG, G_FG,
D_hydrolic_FG,...
    Pr_num_FG, percent_N2_FG,percent_O2_FG,percent_H2O_FG,...
    percent_CO2_FG,Ch_hight);

%% FG - Mass Flow Rate Breakdown
m_dot_FG_H2O = percent_H2O_FG.*m_dot_FG_im;
m_dot_FG_O2 = percent_O2_FG.*m_dot_FG_im;
```

```
m_dot_FG_N2 = percent_N2_FG.*m_dot_FG_im;
m_dot_FG_CO2 =percent_CO2_FG.*m_dot_FG_im;

test=int8(0);
n=int8(0);

T4_im=T3_im;
T7_im=T6_im;
P4_im = P3_im;
P7_im = P6_im;

while (test==0)
    %% CA
    Q_ave_H2O_CA = (19.86/18).*(T4_im-T3_im)-2.*(597/18).*((T4_im+459.7).^...
        0.5-(T3_im+459.7).^0.5)+(7500/18).*(log(T4_im+459.7)-
log(T3_im+459.7));
    Q_ave_N2_CA = (9.47/24).*(T4_im-T3_im) - (3470/24).*(log(T4_im+459.7)...
        -log(T3_im+459.7))+ (1160000/24).*(1/(T4_im+459.7)-1/(T3_im+459.7));
    Q_ave_O2_CA =  (11.515/32).*(T4_im-T3_im)-
2.*(172/32).*((T4_im+459.7).^...
        0.5-(T3_im+459.7).^0.5)+(1530/32).*(log(T4_im+459.7)-
log(T3_im+459.7));
    Q_ave_CO2_CA = (16.2.*(T4_im-T3_im)-6530.*(log(T4_im+459.7)-log(T3_im...
        +459.7))-1410000.*(1/(T4_im+459.7)-1/(T3_im+459.7)))/44;

    Q_ave_CA = [Q_ave_H2O_CA,Q_ave_N2_CA,Q_ave_O2_CA,Q_ave_CO2_CA]*...
        [m_dot_CA_H2O,m_dot_CA_N2,m_dot_CA_O2,m_dot_CA_CO2]';

    %% FG
    Q_ave_H2O_FG = (19.86/18).*(T6_im-T7_im)-2.*(597/18).*((T6_im+459.7).^...
        0.5-(T7_im+459.7).^0.5)+(7500/18).*(log(T6_im+459.7)-
log(T7_im+459.7));
    Q_ave_N2_FG = (9.47/24).*(T6_im-T7_im) - (3470/24).*(log(T6_im+459.7)-...
        log(T7_im+459.7))+ (1160000/24).*(1/(T6_im+459.7)-1/(T7_im+459.7));
    Q_ave_O2_FG =  (11.515/32).*(T6_im-T7_im)-
2.*(172/32).*((T6_im+459.7).^...
        0.5-(T7_im+459.7).^0.5)+(1530/32).*(log(T6_im+459.7)-
log(T7_im+459.7));
    Q_ave_CO2_FG = (16.2.*(T6_im-T7_im)-6530.*(log(T6_im+459.7)-log(T7_im+...
        459.7))-1410000.*(1/(T6_im+459.7)-1/(T7_im+459.7)))/44;

    Q_ave_FG = [Q_ave_H2O_FG,Q_ave_N2_FG,Q_ave_O2_FG,Q_ave_CO2_FG]*...
        [m_dot_FG_H2O,m_dot_FG_N2,m_dot_FG_O2,m_dot_FG_CO2]';

    %% Average Conditions
    T_ave = (T4_im+T3_im+T7_im+T6_im)/4;
    k_steel = -0.000000495314.*T_ave.^2+0.00545322.*T_ave+8.22022;

    r_w = 1/(k_steel/(t_tube/12));


    [~,~, ~, ~, ~, ~,~, ~, h_i_CA_out,~] = Flow_charectoristics
(atm_density,...
        T4_im ,P4_im,m_dot_CA_im,Flow_Area_CA, G_CA, D_hydrolic_CA,
Pr_num_CA, ...
```

```
        percent_N2_CA,percent_O2_CA,percent_H2O_CA, percent_CO2_CA,Ch_hight);


    [~,~, ~, ~, ~, ~,~, ~, h_i_FG_out,~] = Flow_charectoristics (atm_density,
...
        T7_im ,P7_im,m_dot_FG_im,Flow_Area_FG, G_FG, D_hydrolic_FG,
Pr_num_FG,...
        percent_N2_FG,percent_O2_FG,percent_H2O_FG, percent_CO2_FG,Ch_hight);


    U = 1/(2/(h_i_CA_in+h_i_CA_out)+2/(h_i_FG_in+h_i_FG_out)+r_d +r_w);
    if ((T6_im-T4_im)==(T7_im-T3_im))
        LMTD = (T6_im-T4_im);
    else
        LMTD = ((T6_im-T4_im)-(T7_im-T3_im))/log(((T6_im-T4_im)/...
            (T7_im-T3_im)));
    end
    Q_im = U.*A_ht.*LMTD;
    if ((abs(Q_ave_FG-Q_ave_CA)<0.5)&&(abs(Q_im-Q_ave_CA)<0.5)...
            &&(abs(Q_im-Q_ave_FG)<0.5))
        test=1;
    else
        T4_im = T4_im+(Q_im-Q_ave_CA)/1000000;
        T7_im = T7_im-(Q_im-Q_ave_FG)/1000000;

    %% Pressure Drop
    [~,~, ~, Rho_CA_out,Vel_CA_out, ~,~, ~, ~,Delta_P_CA_out] = ...
        Flow_charectoristics (atm_density,T4_im
,P4_im,m_dot_CA_im,Flow_Area_CA,...
        G_CA, D_hydrolic_CA, Pr_num_CA, percent_N2_CA,percent_O2_CA,...
        percent_H2O_CA, percent_CO2_CA,Ch_hight);

    Delta_P_CA = (Delta_P_CA_out+Delta_P_CA_in)/2.*(N_1.*L)+4.*(N_1 -1)*...
        (Rho_CA_in.*Vel_CA_in.^2+Rho_CA_out.*Vel_CA_out.^2)/2/32.2/144;

    P4_im = P3_im - Delta_P_CA;

    [~,~, ~, ~,~, ~,~, ~, ~,Delta_P_FG_out] = Flow_charectoristics ...
        (atm_density,T7_im ,P7_im,m_dot_FG_im,Flow_Area_FG, G_FG,
D_hydrolic_FG,...
        Pr_num_FG,percent_N2_FG,percent_O2_FG,percent_H2O_FG,...
        percent_CO2_FG,Ch_hight);

    Delta_P_FG = (Delta_P_FG_out+Delta_P_FG_in)/2.*N_o;

    P7_im = P6_im - Delta_P_FG;

    end
    n=n+1;

end


%% Conversion
```

```
T4 = (T4_im-32).*5/9+273.15; %[K]
P4 = (P4_im+14.696)/0.145037738; %[KPa]

T7 = (T7_im-32).*5/9+273.15; %[K]
P7 = (P7_im+14.696)/0.145037738; %[KPa]
Q = Q_im/3412.14;%[KW]

if ((isreal(T4)==0)||(isreal(T7)==0)||isreal(P4)==0||(isreal(P7)==0))
    T4 = T3+200;
    T7 = T6-200;
    P4=P3;
    P7=P6;
end
else
    F_S =0;
end
%% Flow Characteristics
function [Visc, Cp, k, density, velocity, Reynolds_num, Colburn_factor, ...
    Friction_factor, h, delta_P] = Flow_charectoristics (atm_density, ...
    T_im ,P_im,m_dot_im,Area, G, D_hydrolic, Pr_num, percent_N2, ...
    percent_O2,percent_H2O, percent_CO2,Ch_hight)

Visc = 2.42.*exp(0.075484.*log(T_im).^2-0.578461.*log(T_im)-2.931705);
%[lb/ft-hr]

Cp= percent_N2.*(9.47-3470/(T_im+459.7)+1160000/(T_im+459.7).^2)/24 ...
    + percent_O2.*(11.515-172/(T_im+459.7).^0.5+1530/(T_im+459.7))/32 ...
    + percent_H2O.*(19.86-597/(T_im+459.7).^0.5+7500/(T_im+459.7))/18 ...
+ percent_CO2.*(11.515-172/(T_im+459.7).^0.5+1530/(T_im+459.7))/32;
%[Btu/lb/F]

k = Cp.*Visc/Pr_num; %[Btu/hr-ft*F]

density = atm_density.*((P_im + 14.696)/14.696)/((T_im+459.63)/519.63);
%[lb/ft^3]
velocity = m_dot_im/density/Area/3600; %[ft/sec]
Reynolds_num = (D_hydrolic/12).* G/Visc;

if (percent_CO2==0)
    if (Reynolds_num<2000)
       Colburn_factor = exp(0.3543232.*log(Reynolds_num)-1.0953035);% j_h
    else
       Colburn_factor = exp(0.0219973.*log(Reynolds_num).^3-...
           0.74268.*log(Reynolds_num).^2+9.0828322.*log(Reynolds_num)...
           -34.1798896);% j_h
    end
else
    Colburn_factor = exp(0.5358039.*log(Reynolds_num)-0.9033314);% j_h
end
if (percent_CO2==0)
   if (Reynolds_num<1000)
        Friction_factor= exp(-0.9964398.*log(Reynolds_num)+...
            8.4994719)/10000;% f[ft^2/in^2]
    else
        Friction_factor= exp(-0.2601568.*log(Reynolds_num)+...
```

```
            3.4073562)/10000;% f[ft^2/in^2]
    end
else
    Friction_factor =exp(-0.1876798.*log(Reynolds_num) + 4.7805861 )/10000;
end

h = Colburn_factor.*k/(D_hydrolic/12).*Pr_num.^(1/3);
% h_i=(j_h*k/D)*Pr^1/3 [Btu/hr F ft^2]

if (percent_CO2==0)
    delta_P = Friction_factor.*G^2/(2.*(32.2.*3600^2).*density.*...
        D_hydrolic/12);
else
    delta_P = Friction_factor.*G^2.*Ch_hight/12/(2.*(32.2.*3600^2).*...
        density.*D_hydrolic/12);
end
```

TURBINE Function

```
function [h_m_m4, s_m_m4, v_m_m4,P5_iso,T5_iso,h_m_m5_iso, ...
    s_m_m5_iso, v_m_m5_iso,P5,T5,h_m_m5, s_m_m5,
v_m_m5,Work_dot,X_water_5,F_S] = ...
    Turbine_G(T4,P4,m_dot_a,m_dot_w, P_amb,X_water_4,F_S)%[K,Pa,Kg/sec]

if (F_S==1)% F_S - Fail safe controller
%% Definitions
eff = 0.8; % Turbine thermal efficiency

M_w  =18.01528; % kg/kmol, assume pure water.
M_a = 28.95849; % kg/kmol  assume 75.5% Nitrogen, 24.5% Oxygen

MR = M_w/M_a; % Molar Ratio

m_dot = m_dot_w + m_dot_a; %[Kg/sec] - Conservation of mass.

MF = m_dot_w/m_dot_a;

x_w=(MF/MR)/((MF/MR)+1);
x_a=1/((MF/MR)+1);

M_m=x_a.*M_a+x_w.*M_w;%[Kg/Kmol]

%% Thermodynamics @ Turbine Inlet

[v_m4, h_m4, s_m4]=Thermodynamic_Properties_G1 (T4, P4, x_a, x_w,X_water_4);
%[m^3/Kmol, KJ/Kmol, KJ/Kmol/K]

v_m_m4 = v_m4/M_m; %[m^3/Kg]
h_m_m4 = h_m4/M_m; %[KJ/Kg]
s_m_m4 = s_m4/M_m; %[KJ/Kg/K]

P5 = ((P_amb.*0.000145037738)+0.5)/0.145037738; %[KPa]

%% Thermodynamics @ Turbine Outlet -Isothermal


T5_iso = T4;
n=0;

    [~, ~, s_m5_iso]=Thermodynamic_Properties_G1 (T5_iso, ...
        P5, x_a, x_w);
    diff = (s_m5_iso-s_m4)/s_m4;

    T_phase_5= Water_Vapor_Saturation_Temp(x_w.*P5.*1000);


    T_top = T5_iso;
    T_bottom = 273.15;
```

```
    while (abs(diff)>0.001)
        n=n+1;
        T_mid = (T_top -T_bottom)/2 + T_bottom;


        if ((T5_iso -T_phase_5)/T5_iso>0.0001)
               X_water_5 = 1;

        elseif((T5_iso -T_phase_5)/T5_iso<-0.0001)

               X_water_5=0;

        else X_water_5 =inf();
        end

        [~, ~, s_m_mid]=Thermodynamic_Properties_G1 (T_mid, ...
        P5, x_a, x_w,X_water_5);
        diff = (s_m4-s_m_mid)/s_m_mid;
        if(n ==3000)
        diff =0;
        end
        if (abs(diff)<0.001)
            T5_iso = T_mid;
        elseif((s_m_mid-s_m4)<0)
            T_bottom = T_mid;
        else
            T_top = T_mid;
        end
    end
[v_m5_iso, h_m5_iso, s_m5_iso]=Thermodynamic_Properties_G1 (T5_iso, ...
        P5, x_a, x_w, X_water_5);


        if (n ==3000)
            F_S=0;
            msgbox('Did not converge','Turbine - Error');
        end


end


P5_iso = P5;%[Pa]
v_m_m5_iso = v_m5_iso/M_m; %[m^3/kg]
h_m_m5_iso = h_m5_iso/M_m; %[KJ/Kg]
s_m_m5_iso = s_m5_iso/M_m; %[KJ/Kg/K]

%% Thermodynamics @ Turbine Outlet - Real
h_m5_eff=h_m4-(eff.*(h_m4-h_m5_iso));

n=0;
T5= T5_iso;

    [~, h_m5, ~]=Thermodynamic_Properties_G1 (T5, P5, x_a, x_w, X_water_5);
```

```
    diff = (h_m5_eff-h_m5)/h_m5_eff;

    T_top = T5+200;
    T_bottom = 273.15;

    while (abs(diff)>0.001)
        n=n+1;
        T_mid = (T_top -T_bottom)/2 + T_bottom;

        [~, h_m_mid, ~]=Thermodynamic_Properties_G1 (T_mid, ...
        P5, x_a, x_w, X_water_5);
        diff = (h_m5_eff-h_m_mid)/h_m5_eff;
        if(n ==3000)
        diff =0;
        end
        if (abs(diff)<0.001)
            T5 = T_mid;
        elseif((h_m_mid-h_m5_eff)<0)
            T_bottom = T_mid;
        else
            T_top = T_mid;
        end
    end
    [v_m5, h_m5, s_m5]=Thermodynamic_Properties_G1 (T5, P5, x_a, x_w,
X_water_5);

    if (n ==3000)
        F_S=0;
        msgbox('Did not converge','Turbine - Error');
    end



v_m_m5 = v_m5/M_m; %[m^3/kg]
h_m_m5 = h_m5/M_m; %[KJ/Kg]
s_m_m5 = s_m5/M_m; %[KJ/Kg/K]

Work_dot = (h_m_m5 - h_m_m4).*m_dot;%[KJ/sec]

end
```

COMBUSTER function

```
function
[T6,P6,m_dot6_H2O,m_dot6_CO2,m_dot6_N2,m_dot6_O2,Q_thro,Q_created,Q_total,F_S
] = ...
Combustor_G(T_amb,P5,T5,m_dot5_w,m_dot5_a,T_Fuel,m_dot_Fuel,F_S)
%Combustor_G(P5,T5,m_dot5_w,m_dot5_a,T_Fuel,m_dot_Fuel,F_S)
% maximum allowable temperature.
%T_max = 1350; %[F]

if (F_S==1)% F_S - Fail safe controller


%% Conversion
T_amb_im = (T_amb-273.15).*9/5+32; %[F]

T5_im = (T5-273.15).*9/5+32; %[F]

T_Fuel_im = (T_Fuel-273.15).*9/5+32; %[F]


%% Turbine Exit Flow


ratio_TE = m_dot5_w/m_dot5_a;
%%% Percent Ratio of TE Mass.

percent_H2O_TE = ratio_TE/(1+ratio_TE);
% _Assume air consist of 24.5% O_2(Oxigen) and 75.5% N_2(Nitrogen) by
% weight._
percent_O2_TE = (1-percent_H2O_TE).*0.245;
percent_N2_TE = (1-percent_H2O_TE).*0.755;
percent_CO2_TE = 0;
%%% Turbine Exit Mass parameters
m_dot5_H2O_im = m_dot5_w.*(1/0.45359237).*3600; %[lb/hr]
m_dot5_N2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*percent_N2_TE;
m_dot5_O2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*percent_O2_TE;
m_dot5_CO2_im = ((m_dot5_a+m_dot5_w).*(1/0.45359237).*3600).*...
    percent_CO2_TE;

%% Fuel Flow

m_dot_Fuel_im = m_dot_Fuel.*(1/0.45359237).*3600; %[lb/hr]
% Combustor efficiency is assumed to be complete.
% Heat losses of the system and combustor will be
% calculated independently.

q_fuel = 22000; %[BTU/lb] HHV of Methane CH4

Q_fuel_im = q_fuel.*m_dot_Fuel_im; %[BTU/hr]
%Q_fuel = Q_fuel_im/3412.14;%[KW]

T_min = T5_im;
```

```
T_max = 10000;
T6_im = T_min+(T_max-T_min)/2;
T_ave = T6_im;
test = 1;
 n=0;
while (test == 1)
    if (n==3000)
        test =0;
        msgbox('Did not converge','Combuster - Error');
        F_S =0;
    else
        n=n+1;
    end
    %% Product Characteristics

    m_dot6_CO2_im =
((m_dot_Fuel/16.042).*(12.0107+2.*15.9994)).*(1/0.45359237).*3600;%[lb/hr];
    m_dot6_N2_im = m_dot5_N2_im;
    m_dot6_H2O_im =
((m_dot_Fuel/16.042).*(4.*1.00794+2.*15.9994).*(1/0.45359237).*3600)+ ...
       m_dot5_H2O_im;%[lb/hr]
    m_dot6_O2_im = m_dot5_O2_im -
((m_dot_Fuel/16.042).*4.*15.9994.*(1/0.45359237).*3600);

    [SpHt_H2O_5, SpHt_H2O_6, SpHt_N2_5,SpHt_N2_6,...
    SpHt_O2_5,SpHt_O2_6,SpHt_CO2_5 ,SpHt_CO2_6]...
    = Q_Average (T5_im, T6_im);

Q_throu_im = ...
    (SpHt_H2O_6.*(T6_im+459.7)-SpHt_H2O_5.*(T5_im+459.7)).*m_dot6_H2O_im...
   +(SpHt_N2_6.*(T6_im+459.7)-SpHt_N2_5.*(T5_im+459.7)).*m_dot6_N2_im...
   +(SpHt_O2_6.*(T6_im+459.7)-SpHt_O2_5.*(T5_im+459.7)).*m_dot6_O2_im...
   +(SpHt_CO2_6.*(T6_im+459.7)-SpHt_CO2_5.*(T5_im+459.7)).*m_dot6_CO2_im;



    %% Created Temperature

    T_CO2_im = (T_Fuel_im.*12.0107 + T5_im.*2.*15.9994)/44.010;
    T_N2_im = T5_im;
    T_O2_im = T5_im;
    T_H2O_im = (T_Fuel_im.*2.*1.00794 + T5_im*15.9994)/(15.9994+2.*1.00794);


SpHt_H2O_FG_In = (19.86/18)-(597/18)/...
        (((T_H2O_im+459.7).^0.5))+(7500/18)/(T_H2O_im+459.7);% [BTU/lb/F]
SpHt_H2O_FG_Out = (19.86/18)-(597/18)/...
        (((T6_im+459.7).^0.5))+(7500/18)/(T6_im+459.7);

SpHt_N2_FG_In = (9.47/24) - (3470/24)/(T_N2_im+459.7)+
(1160000/24)/(T_N2_im+459.7).^2;
SpHt_N2_FG_Out = (9.47/24) - (3470/24)/(T6_im+459.7)+
(1160000/24)/(T6_im+459.7).^2;
```

```
SpHt_O2_FG_In =  (11.515/32)-
(172/32)/(((T_O2_im+459.7).^0.5))+(1530/32)/(T_O2_im+459.7);
SpHt_O2_FG_Out =  (11.515/32)-
(172/32)/(((T6_im+459.7).^0.5))+(1530/32)/(T6_im+459.7);

SpHt_CO2_FG_In = (16.2-6530/(T_CO2_im+459.7)+1410000/(T_CO2_im+459.7).^2)/44;
SpHt_CO2_FG_Out = (16.2-6530/(T6_im+459.7)+1410000/(T6_im+459.7).^2)/44;


    m_dot_CO2_im_cr =
(m_dot_Fuel/16.042).*(12.0107+2.*15.9994).*(1/0.45359237).*3600;%[lb/hr];
    m_dot_N2_im_cr = 0;
    m_dot_H2O_im_cr =
(m_dot_Fuel/16.042).*(4.*1.00794+2.*15.9994).*(1/0.45359237).*3600;%[lb/hr]
    m_dot_O2_im_cr = -(m_dot_Fuel/16.042).*4.*15.9994.*(1/0.45359237).*3600;


Q_created_im = ...
    (SpHt_H2O_FG_Out.*(T6_im+459.7)-
SpHt_H2O_FG_In.*(T_H2O_im+459.7)).*m_dot_H2O_im_cr...
   +(SpHt_N2_FG_Out.*(T6_im+459.7)-
SpHt_N2_FG_In.*(T_N2_im+459.7)).*m_dot_N2_im_cr...
   +(SpHt_O2_FG_Out.*(T6_im+459.7)-
SpHt_O2_FG_In.*(T_O2_im+459.7)).*m_dot_O2_im_cr...
   +(SpHt_CO2_FG_Out.*(T6_im+459.7)-
SpHt_CO2_FG_In.*(T_CO2_im+459.7)).*m_dot_CO2_im_cr;


    Q_total_im = (Q_throu_im + Q_created_im);

    diff = (Q_fuel_im - Q_total_im)/Q_fuel_im;

if (abs(diff)>0.001)

    if (diff>0)
        T_min = T_ave;
        T6_im = T_min+(T_max-T_min)/2;
        T_ave = T6_im;
    else
        T_max = T_ave;
        T6_im = T_min+(T_max-T_min)/2;
        T_ave = T6_im;
    end
else
    test=0;
end

end
P6 = P5;
    m_dot6_CO2 = m_dot6_CO2_im.*0.45359237/3600;
    m_dot6_N2= m_dot6_N2_im.*0.45359237/3600;
    m_dot6_H2O = m_dot6_H2O_im.*0.45359237/3600;
    m_dot6_O2 = m_dot6_O2_im.*0.45359237/3600;
    T6 =(T6_im-32).*5/9+273.15; %[K]
```

```
 Q_thro = (Q_throu_im)/3412.14;%[KW]
 Q_total = (Q_total_im/3412.14); %[KW,KJ/sec]
 Q_created = Q_created_im/3412.14;%[KW]
end
function [SpHt_H2O_FG_Low, SpHt_H2O_FG_Hi, SpHt_N2_FG_Low,SpHt_N2_FG_Hi,...
    SpHt_O2_FG_Low,SpHt_O2_FG_Hi,SpHt_CO2_FG_Low ,SpHt_CO2_FG_Hi]...
    =Q_Average (T_Low, T_Hi)

SpHt_H2O_FG_Low = (19.86/18)-(597/18)/...
        (((T_Low+459.7).^0.5))+(7500/18)/(T_Low+459.7);% [BTU/lb/F]
SpHt_H2O_FG_Hi = (19.86/18)-(597/18)/...
        (((T_Hi+459.7).^0.5))+(7500/18)/(T_Hi+459.7);

SpHt_N2_FG_Low = (9.47/24) - (3470/24)/(T_Low+459.7)+
(1160000/24)/(T_Low+459.7).^2;
SpHt_N2_FG_Hi = (9.47/24) - (3470/24)/(T_Hi+459.7)+
(1160000/24)/(T_Hi+459.7).^2;

SpHt_O2_FG_Low =  (11.515/32)-
(172/32)/(((T_Low+459.7).^0.5))+(1530/32)/(T_Low+459.7);
SpHt_O2_FG_Hi =  (11.515/32)-
(172/32)/(((T_Hi+459.7).^0.5))+(1530/32)/(T_Hi+459.7);

SpHt_CO2_FG_Low = (16.2-6530/(T_Low+459.7)+1410000/(T_Low+459.7).^2)/44;
SpHt_CO2_FG_Hi = (16.2-6530/(T_Hi+459.7)+1410000/(T_Hi+459.7).^2)/44;
```

Thermodynamic Properties Function

```
function [v_m, h_m, s_m]=Thermodynamic_Properties_G1 (T, P, ~ , x_w, X_water)
% T [K] , P [KPA]

% Ref 3 -S. Herrmann, H.-J. Kretzschmar, V. Teske, E. Vogel, P. Ulbig,
    %R. Span, D.P. Gatley, "Determination of Thermodynamic and Transport
    %Properties of Humid Air for Power-Cycle Calculations"[Online]. PTB,
    %Braunschweig, Germany, January 2009. Available:
    %http://www.thermodynamics-zittau.de/ [11 January 2011]

 % Psi_w - molar fraction of water in humid air.
 % Psi_a - molar fraction of dry air in humid air.
 % Epsilon = Mole_w/Mole_a=R_tild_a/R_tild_w
 % x_w = m_w/m_a <- absolute humidity, mass of water/mass of dry air
    % CONFIRM IN COMP & SPRAY1
 %X_water = % of liquid vapor in a phase changing fluid.
 % if X_water = 0 - the fluid is 100% liquid
 % if X_water = 1 - The Fluid is 100% steam

M_w = 18.015208; %kg/kmol
M_a = 28.9586; %kg/kmol
Psi_w = x_w/(x_w + (M_w/M_a));
 Psi_a = 1 - Psi_w;



[~,~,Rho_a,h_a,~,s_a]=Property_Iterator_G1('air',P,T);

h_a_0 = 7914.441; %KJ/Kmol @ T = 273.16 & P = 101.325 KPa
s_a_0 = 196.45; %%KJ/Kmol K @ T = 273.16 & P = 101.325 KPa


D_h_a = h_a - h_a_0;
D_s_a = s_a - s_a_0;

P_w = P.*Psi_w;
%%

if (X_water<1 && X_water >0)

    T_sat = Water_Vapor_Saturation_Temp(P_w*1000);

    [~,~,~,h_w_s,~,s_w_s]=Property_Iterator_G1('water',P_w,(T_sat+0.1));
    [~,~,~,h_w_l,~,s_w_l]=Property_Iterator_G1('water',P_w,(T_sat-0.1));

    h_w = h_w_l + X_water.*(h_w_s-h_w_l);
    s_w = s_w_l + X_water.*(s_w_s-s_w_l);

else
    [~,~,~,h_w,~,s_w]=Property_Iterator_G1('water',P_w,T);

end
```

```
h_w_0  = 0.771244; %  %KJ/Kmol @ T = 273.17 & P = 0.612 KPa
s_w_0 = 0.002782;  %KJ/Kmol K @ T = 273.17 & P = 0.612 KPa

D_h_w = h_w - h_w_0;
D_s_w = s_w - s_w_0;



v_m = 1/Rho_a;

h_m = Psi_a.*D_h_a+Psi_w.*D_h_w; % in HBC Dissociation of Temperature affects
are not observed.

DELTA_S_irr = -8.314472.*(Psi_a.*log(Psi_a)+Psi_w.*log(Psi_w));

s_m = Psi_w.*D_s_w+Psi_a.*D_s_a+DELTA_S_irr;
end
```

Property Iterator Function

```
%% Property Iterator

% the purpose of this function is to conduct iteration of the property
% function and determine thermodynaic results.

function [P,T,Rho,h,u,s]=Property_Iterator_G1(fun,P,T)

%global M_size P_range_s P_range_l T_range Rho_range_l Rho_range_s

    % fun <= air or water
    % P <= # KPa
    % T <= # K
    % u <= # j/mol
    % h <= # j/mol
    % s <= # j/(mol K)

    fun=lower(fun);

switch fun
case 'air'

    clear Rho_top Rho_bottom Rho_middle

count = 0;

% design to work on Gas Air, for liquid air denisty bounderies will require
% adjustment.
Rho_bottom = 0.00001; %kmol/m^3 <- Start value for density.
Rho_top = 1;
Boundary_ok = boolean(false);
found = boolean(false);
while (~Boundary_ok)

        [P_a_top, ~,~,~,~,~] = Air_properties_G(T, Rho_top);
        [P_a_bottom, ~,~,~,~,~] = Air_properties_G(T, Rho_bottom);

        if (P_a_top < 0)
            Rho_top = Rho_top .*2;

        elseif (P_a_top < P)
            Rho_bottom = Rho_top; % replace bottom boundary with Top.
            Rho_top = Rho_top.*1.5; % Inclrease top boundary by 50%.

        elseif (P_a_bottom > P)
            Rho_top = Rho_bottom;
            Rho_bottom = Rho_bottom.*0.5;
        else
            Boundary_ok =true;
        end
end
```

```
while (~found && (count < 3001)) %found is false = Not found is true.

        [P_a_top, ~,~,~,~,~] = Air_properties_G(T, Rho_top);
        [P_a_bottom, ~,~,~,~,~] = Air_properties_G(T, Rho_bottom);

        if (P_a_top<P)
            Rho_top=Rho_top.*1.05;
            [P_a_top, ~,~,~,~,~] = Air_properties_G(T, Rho_top);
        elseif (P_a_bottom>P)
            Rho_bottom=Rho_bottom.*0.95;
            [P_a_bottom, ~,~,~,~,~] = Air_properties_G(T, Rho_bottom);
        end

        if (abs(P_a_top - P)/P <= 0.001)
            Rho_a = Rho_top;
            found = true;

        elseif(abs(P_a_bottom - P)/P <= 0.001)
                Rho_a = Rho_bottom;
                found = true;

        else
            Rho_middle = Rho_bottom + (Rho_top - Rho_bottom)/2;
            [P_a_middle, ~,~,~,~, ~] = Air_properties_G(T, Rho_middle);

            if (abs(P_a_middle - P)/P <= 0.001)
                Rho_a = Rho_middle;
                found = 1;

            elseif ((P- P_a_bottom) <(P_a_top - P))

                Rho_top = Rho_middle;

            else
                Rho_bottom = Rho_middle;
            end
        end
        count = count + 1;

end

[P, ~, ~, u, h, ...
    s]=Air_properties_G(T, Rho_a);
Rho = Rho_a;


case 'water'

P_w = P/1000;%[MPa]

v = IAPWS_IF97('v_pT',P_w,T);% [Kg/m^3] third party open source code.
Rho_w = (1/v)/18.015;
[~, ~, ~, u, h, s] = Water_properties_G(T, Rho_w);
```

```
Rho =Rho_w;
end
end
```

Air Properties Function

```
function [P_a, c_p_a, c_v_a, u_a, h_a, ...
    s_a]=Air_properties_G(T_a, Rho_a)

%% Air properites
% Ref 1 - Lemmon, E. W.; Jacobsen, R. T.; Penoncello, S. G.; Friend, D. G.:
% Thermodynamic Properties of Air and Mixtures of Nitrogen, Argon, and
% Oxygen from 60 to 2000 K at Pressures to 2000 MPa. J. Phys. Chem. Ref.
% Data 29, 331-385 (2000).

% Constants
M_a  = 28.9586; % kg/kmol
R_a_tild = 0.287117.^10; %KJ/(Kg*K)
R_lem = 8.314510; %KJ/(Kmol*K)
T_j_a = 132.6312; %K
Rho_j_a = 10.4477; %Kmol/m^3
digits (15);
% Variables
% -> T_a, Rho_a

%% Governing Formula
% a_a(Rho_a,T)/(R_lem*T) = a_a(Delta_a,Tau_a) =a_a_0(Delta_a,Tau_a)+...
% a_a_r(Delta_a,Tau_a)

% a_a_0(Delta_a,Tau_a) - Ideal gas contribution
% a_a_r(Delta_a,Tau_a) - Residual contribution

Delta_a = Rho_a/Rho_j_a;
Tau_a = T_j_a/T_a;
%% Coefficents

% Define Coefficent matrix for Alpha_0 of air(Ref 1 - Table 12)
Coeff_a=zeros(13,1);
 Coeff_a(:,1) =[   0.605719400.*10^-7; - 0.210274769.*10^-4;
                 - 0.158860716.*10^-3; -13.841928076       ;
                  17.275266575       ; - 0.195363420.*10^-3;
                   2.490888032       ;   0.791309509       ;
                   0.212236768       ; - 0.197938904       ;
                  25.36365           ;  16.90741           ;
                  87.31279           ] ;

% Define Coefficent matrix (Ref 1 - Table 13)
Coeff=zeros(19,4);

% Populate Coefficient matrix
    % N_k values - Column 1
    Coeff(:,1) =[ 0.118160747229       ; 0.713116392079       ;
                 -0.161824192067.*10^ 1; 0.714140178971.*10^-1;
                 -0.865421396646.*10^-1; 0.134211176704       ;
                  0.112626704218.*10^-1;-0.420533228842.*10^-1;
                  0.349008431982.*10^-1; 0.164957183186.*10^-3;
                 -0.101365037912       ;-0.173813690970       ;
                 -0.472103183731.*10^-1;-0.122523554253.*10^-1;
                 -0.146629609713       ;-0.316055879821.*10^-1;
```

```
                 0.233594806142.*10^-3; 0.148287891978.*10^-1;
                -0.938782884667.*10^-2];

    % i_k values - Column 2
    Coeff(:,2) = [ 1;  1; 1; 2; 3;
                   3;  4; 4; 4; 6;
                   1;  3; 5; 6; 1;
                   3; 11; 1; 3];

    % j_k values - Column 3
    Coeff(:,3) = [    0; 0.33; 1.01;    0;    0;
                   0.15;    0;  0.2; 0.35; 1.35;
                    1.6;  0.8; 0.95; 1.25;  3.6;
                      6; 3.25;  3.5;   15];

    % l_k values - Column 4
    Coeff(:,4)  =[0; 0; 0; 0; 0;
                  0; 0; 0; 0; 0;
                  1; 1; 1; 1; 2;
                  2; 2; 3; 3];
%% Ideal Gas Component of Equation of State for Air
    % Sum component for a_a_0
            N = 0;
            for i = 1:5
                N_new = Coeff_a(i,1)*Tau_a.^(i-4);
                N = N + N_new;
            end
    a_a_0 = log(Delta_a)+ N + Coeff_a(6,1).*Tau_a.^(1.5) + ...
            Coeff_a(7,1).*log(Tau_a)+ ...
            Coeff_a(8,1).*log(1-exp(-Coeff_a(11,1).*Tau_a))+...
            Coeff_a(9,1).*log(1-exp(-Coeff_a(12,1).*Tau_a))+...
            Coeff_a(10,1).*log(2/3+exp(Coeff_a(13,1).*Tau_a));


%% Residual Component of Equation of State for Air
    % Sum Component #1 for a_a_r
       N = 0;
       for k = 1:10
        N_new = Coeff(k,1).*Delta_a.^(Coeff(k,2)).*Tau_a.^(Coeff(k,3));
        N=N+N_new;
       end
       a_a_r_sum1 = N;
    % Sum Component #2 for a_a_r

       N = 0;
       for k = 11:19
            N_new = Coeff(k,1).*Delta_a.^(Coeff(k,2)).*...
            Tau_a.^(Coeff(k,3)).*exp(-Delta_a.^(Coeff(k,4)));
        N=N+N_new;
       end
       a_a_r_sum2 = N;

       a_a_r = a_a_r_sum2+a_a_r_sum1;

%% First, Second and mixed Derivatives for a_a_0
```

```
%    Da_a_0 = 1/Delta_a; <- Not used for this HBC study

%    DDa_a_0 = -1/Delta_a.^2; <- Not used for this HBC study

    %Ta_a_0 Calculations --->
        N = 0;
        for i = 1:5
            N_new = (i-4).*Coeff_a(i,1).*Tau_a.^(i-5);
            N=N+N_new;
        end


    Ta_a_0= N + 1.5.*Coeff_a(6,1).*Tau_a.^(0.5)+Coeff_a(7,1)/Tau_a+...
        Coeff_a(8,1).*Coeff_a(11,1)/(exp(Coeff_a(11,1).*Tau_a)-1)+...
        Coeff_a(9,1).*Coeff_a(12,1)/(exp(Coeff_a(12,1).*Tau_a)-1)+...
        Coeff_a(10,1).*Coeff_a(13,1)/((2/3).*exp(-Coeff_a(13,1).*Tau_a)+1);

    %TTa_a_0 Calculations --->
        N = 0;
        for i = 1:5
            N_new = (i-4).*(i-5).*Coeff_a(i,1).*Tau_a.^(i-6);
            N=N+N_new;
        end


    TTa_a_0 = N + 0.75.*Coeff_a(6,1).*Tau_a.^(-0.5)-Coeff_a(7,1)/Tau_a.^2-...
        (Coeff_a(8,1).*Coeff_a(11,1).^2.*exp(Coeff_a(11,1).*Tau_a))/...
        (exp(Coeff_a(11,1).*Tau_a)-1).^2-...
        (Coeff_a(9,1).*Coeff_a(12,1).^2.*exp(Coeff_a(12,1).*Tau_a))/...
        (exp(Coeff_a(12,1).*Tau_a)-1).^2+...
        ((2/3).*Coeff_a(10,1).*Coeff_a(13,1).^2.*exp(-
Coeff_a(13,1).*Tau_a))/...
        ((2/3).*(exp(-Coeff_a(13,1).*Tau_a)+1)).^2;

%    DTa_a_0 = 0; <- Not used for this HBC study

% NOTE - All partial derivative are noted in this script however not all
    % are used in HBC application. All partial derivative mantioned for
    %completeness of the study.

%% First, Second and mixed Derivatives for a_a_r
%%(Residual Helmholtz Energy Equation)

  % Da_a_r calculation
    % Sum Component #1 for Da_a_r
    N = 0;
        for k = 1:10
    N_new = Coeff(k,2).*Coeff(k,1).*Delta_a.^(Coeff(k,2)-1).*...
        Tau_a.^(Coeff(k,3));
    N=N+N_new;
        end
    Da_a_r_sum1 = N;
    % Sum Component #2 for Da_a_r
```

```
    N = 0;
        for k = 11:19
    N_new = Coeff(k,1).*Delta_a.^(Coeff(k,2)-1).*...
    Tau_a.^(Coeff(k,3)).*...
    exp(-Delta_a.^Coeff(k,4)).*...
    (Coeff(k,2)-Coeff(k,4).*Delta_a.^Coeff(k,4));
    N=N+N_new;
        end
    Da_a_r_sum2 = N;

    Da_a_r = Da_a_r_sum2+Da_a_r_sum1;

% DDa_a_r calculation
    % Sum Component #1 for DDa_a_r
    N = 0;
        for k = 1:10
    N_new = Coeff(k,2).*(Coeff(k,2)-1).*Coeff(k,1).*...
        Delta_a.^(Coeff(k,2)-2).*Tau_a.^(Coeff(k,3));
    N=N+N_new;
        end
    DDa_a_r_sum1 = N;
    % Sum Component #2 for DDa_a_r

    N = 0;
        for k = 11:19
    N_new = Coeff(k,1).*Delta_a.^(Coeff(k,2)-2).*Tau_a.^(Coeff(k,3)).*...
    exp(-Delta_a.^Coeff(k,4)).*...
    ((Coeff(k,2)-Coeff(k,4).*Delta_a.^Coeff(k,4)).*...
    (Coeff(k,2)-1-Coeff(k,4).*Delta_a.^Coeff(k,4))-...
    Coeff(k,4).^2.*Delta_a.^Coeff(k,4));
    N=N+N_new;
        end
    DDa_a_r_sum2 = N;

    DDa_a_r = DDa_a_r_sum2+DDa_a_r_sum1;

    % Ta_a_r calculation
    % Sum Component #1 for Ta_a_r
    N = 0;
        for k = 1:10
    N_new = Coeff(k,3).*Coeff(k,1).*Delta_a.^(Coeff(k,2)).*...
        Tau_a.^(Coeff(k,3)-1);
    N=N+N_new;
        end
    Ta_a_r_sum1 = N;
    % Sum Component #2 for Ta_a_r

    N = 0;
        for k = 11:19
    N_new = Coeff(k,3).*Coeff(k,1).*Delta_a.^(Coeff(k,2)).*...
    Tau_a.^(Coeff(k,3)-1).*...
    exp(-Delta_a.^(Coeff(k,4)));
    N=N+N_new;
        end
```

```
Ta_a_r_sum2 = N;

Ta_a_r = Ta_a_r_sum2+Ta_a_r_sum1;

% TTa_a_r calculation
% Sum Component #1 for TTa_a_r
N = 0;
    for k = 1:10
N_new = (Coeff(k,3)-1).*Coeff(k,3).*Coeff(k,1).*...
    Delta_a.^(Coeff(k,2)).*Tau_a.^(Coeff(k,3)-2);
N=N+N_new;
    end
TTa_a_r_sum1 = N;
% Sum Component #2 for TTa_a_r

N = 0;
    for k = 11:19
N_new = Coeff(k,3).*(Coeff(k,3)-1).*Coeff(k,1).*...
Delta_a.^(Coeff(k,2)).*Tau_a.^(Coeff(k,3)-2).*...
exp(-Delta_a.^(Coeff(k,4)));
N=N+N_new;
    end
TTa_a_r_sum2 = N;

TTa_a_r = TTa_a_r_sum2+TTa_a_r_sum1;

% DTa_a_r calculation
 % Sum Component #1 for DTa_a_r
 N = 0;
     for k = 1:10
 N_new = Coeff(k,2).*Coeff(k,3).*Coeff(k,1).*...
     Delta_a.^(Coeff(k,2)-1).*Tau_a.^(Coeff(k,3)-1);
 N=N+N_new;
     end
 DTa_a_r_sum1 = N;
 % Sum Component #2 for DTa_a_r

 N = 0;
     for k = 11:19
 N_new = Coeff(k,3).*Coeff(k,1).*Delta_a.^(Coeff(k,2)-1).*...
     Tau_a.^(Coeff(k,3)-1).*exp(-Delta_a.^(Coeff(k,4))).*...
     (Coeff(k,2)-Coeff(k,4).*Delta_a.^(Coeff(k,4)));
 N=N+N_new;
     end
 DTa_a_r_sum2 = N;

 DTa_a_r = DTa_a_r_sum2+DTa_a_r_sum1;

 clear N N_new TTa_a_r_sum2 TTa_a_r_sum1 Ta_a_r_sum1 Ta_a_r_sum2...
     a_a_r_sum2 a_a_r_sum1 DTa_a_r_sum2 DTa_a_r_sum1 Da_a_r_sum1 ...
     Da_a_r_sum2 DDa_a_r_sum1 DDa_a_r_sum2;


%% Thermodynamic properties of Air
    % Pressure Calculations
```

```
     P_a = R_lem.*T_a.*Rho_a.*(1+Delta_a.*Da_a_r);

            % Specific value of the property is calculated from Molar value
            %  as follow: x_tild_a = x_a/M_a
            %ie. Rho_tild_a = Rho_a/M_a;

    % Isobaric Heat capacity
c_p_a = R_lem.*(-(Tau_a.^2).*(TTa_a_0+TTa_a_r)+...
    ((1+Delta_a.*Da_a_r-Delta_a.*Tau_a.*DTa_a_r).^2)/...
    (1+2.*Delta_a.*Da_a_r+(Delta_a.^2).*DDa_a_r));

    % Isochoric Heat Capacity
c_v_a = -R_lem.*((Tau_a.^2).*(TTa_a_0+TTa_a_r));

    % Enthalpy
h_a = R_lem.*T_a.*(Tau_a.*(Ta_a_0+Ta_a_r)+Delta_a.*Da_a_r+1);

    % Internal Energy
u_a = R_lem.*T_a.*(Tau_a.*(Ta_a_0+Ta_a_r));

    % Entropy
s_a = R_lem.*(Tau_a.*(Ta_a_0+Ta_a_r)-a_a_0-a_a_r);

% NOTE - This script does not include Speed of Sound and Isentropic
    % Exponent calculation as they are not required for HBC study.

end
```

Water Properties Function

```
function [P_w, c_p_w, c_v_w, u_w, h_w, s_w]=Water_properties_G(T_w, Rho_w)

%% Water Properties
% Ref 2 - Wagner, W.; Pruß, A.: The IAPWS
% Formulation 1995 for the Thermodynamic Properties of Ordinary Water
% Substance for General and Scientific Use. J. Phys.Chem. Ref. Data 31,
% 387-535 (2002).

% Constants
M_w  = 18.015268; % kg/kmol
R_w_tild = 0.46151805; %KJ/(Kg*K)
R_95 = 8.314371.^12; %KJ/(Kmol*K)
T_c_w = 647.096; %K
Rho_c_w = 322.0; %Kg/m^3

% Variables
% -> T_w, Rho_tild_w

%% Governing Formula
% a_w(Rho_tild_w,T)/(R_w_tild*T) = a_w(Delta_w,Tau_w)
=a_w_0(Delta_w,Tau_w)+...
% a_w_r(Delta_w,Tau_w)

% a_w_0(Delta_w,Tau_w) - Ideal gas contribution
% a_w_r(Delta_w,Tau_w) - Residual contribution

Rho_tild_w = Rho_w.*M_w; %ie. Rho_tild_w = Rho_w/M_w
Delta_w = Rho_tild_w/Rho_c_w;
Tau_w = T_c_w/T_w;




%% Ideal Component of Equation of State for Water

% Coefficents
% Define Coefficent matrix for a_w_0(Ref 2 - Table 6.1)

%-> Populate Coefficient matrix
Coeff_w_0 = zeros(8,2);

    % n_i_0 values - Column 1
Coeff_w_0(:,1) = [-8.32044648201;  6.6832105268;  3.00632;  0.012436;...
                   0.97315;        1.27950;       0.96956;  0.24873  ];

    % gamma_i_0 values - Column 1

Coeff_w_0(:,2) = [inf();           inf();           inf();
1.28728967;...
                  3.53734222;  7.74073708;  9.24437796;  27.5075105];
```

```
%-> calculate a_w_0(Delta_w,Tau_w)
 N = 0;
       for i = 4:8
        N_new = Coeff_w_0(i,1).*log(1-exp(-Coeff_w_0(i,2).*Tau_w));
        N=N+N_new;
       end
        a_w_0 = log(Delta_w)+Coeff_w_0(1,1)+Coeff_w_0(2,1).*Tau_w...
            +Coeff_w_0(3,1).*log(Tau_w)+N;

%% First, Second and mixed Derivatives for a_w_0
   % ref table 6.4
  Da_w_0 = 1/Delta_w;

  DDa_w_0 = -1/Delta_w.^2;


  N = 0;
       for i = 4:8
        N_new = Coeff_w_0(i,1).*Coeff_w_0(i,2).*...
            (((1-exp(-Coeff_w_0(i,2).*Tau_w)).^(-1))-1);
        N=N+N_new;
       end
  Ta_w_0 = Coeff_w_0(2,1)+Coeff_w_0(3,1)/Tau_w+N;

  N = 0;
       for i = 4:8
        N_new = Coeff_w_0(i,1).*(Coeff_w_0(i,2).^2).*...
            exp(-Coeff_w_0(i,2).*Tau_w).*...
            ((1-exp(-Coeff_w_0(i,2).*Tau_w)).^-2);
        N=N+N_new;
       end
  TTa_w_0 = -Coeff_w_0(3,1)/Tau_w.^2-N;

  DTa_w_0 = 0;

%% Residual Component of Equation of State for Water
% Coefficents
% Define Coefficent matrix for a_w_r(Ref 2 - Table 6.2)

Coeff_w_r=inf(56,16);

% ->Populate Coefficient matrix
    % a_i values - Column 1
Coeff_w_r(55:56,1) =[3.5; 3.5];

    % b_i values - Column 2
Coeff_w_r(55:56,2) =[0.85; 0.95];

    % c_i values - Column 3
for i=8:22;
    Coeff_w_r(i,3) = 1;
end
for i=23:42;
```

```
    Coeff_w_r(i,3) = 2;
end
    Coeff_w_r(43:51,3) = [3; 3; 3; 3; 4; 6; 6; 6; 6];

     % d_i values - Column 4
Coeff_w_r(:,4) = [ 1;  1; 1;  2;  2;  3;  4; 1; 1;  1;  2;  2; 3; 4; 4;...
                   5;  7; 9; 10; 11; 13; 15; 1; 2;  2;  2;  3; 4; 4; 4;...
                   5;  6; 6;  7;  9;  9;  9; 9; 9; 10; 10; 12; 3; 4; 4;...
                   5; 14; 3;  6;  6;  6;  3; 3; 3;  inf();  inf()];

     % t_i values - Column 5
Coeff_w_r(:,5) = [-0.5; 0.875;  1; 0.5; 0.75; 0.375;  1;  4;  6; 12; ...
                    1;     5;  4;   2;   13;     9;  3;  4; 11;  4; ...
                   13;     1;  7;   1;    9;    10; 10;  3;  7; 10; ...
                   10;     6; 10;  10;    1;     2;  3;  4;  8;  6; ...
                    9;     8; 16;  22;   23;    23; 10; 50; 44; 46; ...
                   50;     0;  1;   4;    inf();     inf()];

    % n_i values - Column 6
Coeff_w_r(:,6) =  [ 0.12533547935523*10^-1 ;  0.78957634722828*10^ 1; ...
                   -0.87803203303561*10^ 1 ;  0.31802509345418      ; ...
                   -0.26145533859358       ; -0.78199751687981*10^-2; ...
                    0.88089493102134*10^-2 ; -0.66856572307965      ; ...
                    0.20433810950965       ; -0.66212605039687*10^-4; ...
                   -0.19232721156002       ; -0.25709043003438      ; ...
                    0.16074868486251       ; -0.40092828925807*10^-1; ...
                    0.39343422603254*10^-6 ; -0.75941377088144*10^-5; ...
                    0.56250979351888*10^-3 ; -0.15608652257135*10^-4; ...
                    0.11537996422951*10^-8 ;  0.36582165144204*10^-6; ...
                   -0.13251180074668*10^-11; -0.62639586912454*10^-9; ...
                   -0.10793600908932       ;  0.17611491008752*10^-1; ...
                    0.22132295167546       ; -0.40247669763528      ; ...
                    0.58083399985759       ;  0.49969146990806*10^-2; ...
                   -0.31358700712549*10^-1 ; -0.74315929710341      ; ...
                    0.47807329915480       ;  0.20527940895948*10^-1; ...
                   -0.13636435110343       ;  0.14180634400617*10^-1; ...
                    0.83326504880713*10^-2 ; -0.29052336009585*10^-1; ...
                    0.38615085574206*10^-1 ; -0.20393486513704*10^-1; ...
                   -0.16554050063734*10^-2 ;  0.19955571979541*10^-2; ...
                    0.15870308324157*10^-3 ; -0.16388568342530*10^-4; ...
                    0.43613615723811*10^-1 ;  0.34994005463765*10^-1; ...
                   -0.76788197844621*10^-1 ;  0.22446277332006*10^-1; ...
                   -0.62689710414685*10^-4 ; -0.55711118565645*10^-9; ...
                   -0.19905718354408       ;  0.31777497330738      ; ...
                   -0.11841182425981       ; -0.31306260323435*10^ 2; ...
                    0.31546140237781*10^ 2 ; -0.25213154341695*10^ 4; ...
                   -0.14874640856724       ;  0.31806110878444     ];

    % A_i values - Column 7
Coeff_w_r(55:56, 7) = [0.32; 0.32];

    % B_i values - Column 8
Coeff_w_r(55:56, 8) = [ 0.2;  0.2];

    % C_i values - Column 9
Coeff_w_r(55:56, 9) = [  28;   32];
```

```
    % D_i values - Column 10
Coeff_w_r(55:56,10) = [ 700;  800];

    % alpha_i values - Column 11
Coeff_w_r(52:54,11) = [20; 20; 20];

    % bravo_i values - Column 12
Coeff_w_r(52:56,12) = [150; 150; 250; 0.3; 0.3];

    % gamma_i values - Column 13
Coeff_w_r(52:54,13) = [1.21; 1.21; 1.25];

    % epsilon_i values - Column 14
Coeff_w_r(52:54,14) = [1; 1; 1];

%-> calculate a_w_r(Delta_w,Tau_w)

Comp1 = 0;
       for i = 1:7
        Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
            Tau_w.^Coeff_w_r(i,5);
        Comp1=Comp1+Comp;
       end

Comp2 = 0;
       for i = 8:51
        Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
            Tau_w.^Coeff_w_r(i,5).*exp(-Delta_w.^Coeff_w_r(i,3));
        Comp2=Comp2+Comp;
       end

Comp3 = 0;
       for i = 52:54
        Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
            Tau_w.^Coeff_w_r(i,5).*exp(-Coeff_w_r(i,11).*...
            (Delta_w-Coeff_w_r(i,14)).^2-Coeff_w_r(i,12).*...
            (Tau_w-Coeff_w_r(i,13)).^2);

        Comp3=Comp3+Comp;
       end

Comp4 = 0;
       for i = 55:56
        psi = exp(-Coeff_w_r(i,9).*((Delta_w-1).^2)-...
                Coeff_w_r(i,10).*(Tau_w-1).^2);
        theta =(1-Tau_w)+Coeff_w_r(i,7).*((Delta_w-1).^2).^...
                (1/(2.*Coeff_w_r(i,12)));
        DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                Coeff_w_r(i,1));

       Comp = Coeff_w_r(i,6).*DELTA.^(Coeff_w_r(i,2)).*Delta_w.*psi;

        Comp4=Comp4+Comp;
```

```
        end

    a_w_r = Comp1+Comp2+Comp3+Comp4;



%% First, Second and mixed Derivatives for a_w_r

    % ref table 6.5
% Da_w_r
Comp1 = 0;
        for i = 1:7
         Comp = Coeff_w_r(i,6).*Coeff_w_r(i,4).*...
             Delta_w.^(Coeff_w_r(i,4)-1).*Tau_w.^(Coeff_w_r(i,5));
         Comp1=Comp1+Comp;
        end

Comp2 = 0;
        for i = 8:51
         Comp = Coeff_w_r(i,6).*exp(-Delta_w.^Coeff_w_r(i,3)).*...
             (Delta_w.^(Coeff_w_r(i,4)-1).*Tau_w.^(Coeff_w_r(i,5)).*...
             (Coeff_w_r(i,4)-Coeff_w_r(i,3).*Delta_w.^Coeff_w_r(i,3)));

         Comp2=Comp2+Comp;
        end

Comp3 = 0;
        for i = 52:54
         Comp = Coeff_w_r(i,6).*Delta_w.^(Coeff_w_r(i,4)).*...
             Tau_w.^(Coeff_w_r(i,5)).*exp(-Coeff_w_r(i,11).*...
             (Delta_w-Coeff_w_r(i,14)).^2-Coeff_w_r(i,12).*...
             (Tau_w-Coeff_w_r(i,13)).^2).*...
             (Coeff_w_r(i,4)/Delta_w-2.*Coeff_w_r(i,11).*...
             (Delta_w-Coeff_w_r(i,14)));

         Comp3=Comp3+Comp;
        end

Comp4 = 0;
        for i = 55:56
         psi = exp(-Coeff_w_r(i,9).*((Delta_w-1).^2)-...
                 Coeff_w_r(i,10).*(Tau_w-1).^2);
         theta =(1-Tau_w)+Coeff_w_r(i,7).*((Delta_w-1).^2).^...
                 (1/(2.*Coeff_w_r(i,12)));
         DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                 Coeff_w_r(i,1));
         D_DELTA = (Delta_w-1).*(Coeff_w_r(i,7).*theta.*2/...
             Coeff_w_r(i,12).*((Delta_w-1).^2).^...
             (1/(2.*Coeff_w_r(i,12))-1)+2.*Coeff_w_r(i,8).*...
             Coeff_w_r(i,1).*((Delta_w-1).^2).^(Coeff_w_r(i,1)-1));

         D_DELTA_b_i = Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1).* D_DELTA;

        D_psi = -2.*Coeff_w_r(i,9).*(Delta_w-1).*psi;
```

```
      Comp = Coeff_w_r(i,6).*(DELTA.^Coeff_w_r(i,2).*(psi+...
          Delta_w.*D_psi)+D_DELTA_b_i.*D_psi);

      Comp4 = Comp4+Comp;
      end
 Da_w_r = Comp1+Comp2+Comp3+Comp4;

 %DDa_w_r
Comp1=0;
Comp2=0;
Comp3=0;
Comp4=0;

% Comp1
      for i = 1:7
       Comp = Coeff_w_r(i,6).*Coeff_w_r(i,4).*(Coeff_w_r(i,4)-1).*...
           Delta_w.^(Coeff_w_r(i,4)-2).*Tau_w.^(Coeff_w_r(i,5));
       Comp1=Comp1+Comp;
      end

% Comp2
      for i = 8:51
       Comp = Coeff_w_r(i,6).*exp(-Delta_w.^Coeff_w_r(i,3)).*...
           (Delta_w.^(Coeff_w_r(i,4)-2).*Tau_w.^(Coeff_w_r(i,5)).*...
           ((Coeff_w_r(i,4)-Coeff_w_r(i,3).*Delta_w.^Coeff_w_r(i,3)).*...
           (Coeff_w_r(i,4)-1-Coeff_w_r(i,3).*Delta_w.^Coeff_w_r(i,3))-...
           Coeff_w_r(i,3).^2.*Delta_w.^Coeff_w_r(i,3)));

       Comp2=Comp2+Comp;
      end

% Comp3

      for i = 52:54
       Comp = Coeff_w_r(i,6).*Tau_w.^(Coeff_w_r(i,5)).*...
           exp(-Coeff_w_r(i,11).*(Delta_w-Coeff_w_r(i,14)).^2-...
           Coeff_w_r(i,12).*(Tau_w-Coeff_w_r(i,13)).^2).*...
           (-2.*Coeff_w_r(i,11).*Delta_w.^Coeff_w_r(i,4)+4.*...
           Coeff_w_r(i,11).^2.*Delta_w.^Coeff_w_r(i,4).*...
           (Delta_w-Coeff_w_r(i,14)).^2-4.*Coeff_w_r(i,4).*...
           Coeff_w_r(i,11).*Delta_w.^(Coeff_w_r(i,4)-1).*...
           (Delta_w-Coeff_w_r(i,14))+Coeff_w_r(i,4).*...
           (Coeff_w_r(i,4)-1).*Delta_w.^(Coeff_w_r(i,4)-2));


       Comp3=Comp3+Comp;
      end

% Comp4
      for i = 55:56
       psi = exp(-Coeff_w_r(i,9).*((Delta_w-1).^2)-...
               Coeff_w_r(i,10).*(Tau_w-1).^2);
       theta =(1-Tau_w)+Coeff_w_r(i,7).*((Delta_w-1).^2).^...
               (1/(2.*Coeff_w_r(i,12)));
```

```
        DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                Coeff_w_r(i,1));
        D_DELTA = (Delta_w-1).*(Coeff_w_r(i,7).*theta.*2/...
            Coeff_w_r(i,12).*((Delta_w-1).^2).^...
            (1/(2.*Coeff_w_r(i,12))-1)+2.*Coeff_w_r(i,8).*...
            Coeff_w_r(i,1).*((Delta_w-1).^2).^(Coeff_w_r(i,1)-1));

        D_DELTA_b_i = Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1).* D_DELTA;
      % End here
        DD_DELTA  = (1/(Delta_w-1).*D_DELTA)+(Delta_w-1).^2.*...
        (4.*Coeff_w_r(i,8).*Coeff_w_r(i,1).*(Coeff_w_r(i,1)-1).*...
          ((Delta_w-1).^2).^(Coeff_w_r(i,1)-2)+2.*Coeff_w_r(i,7).^2.*...
          (1/Coeff_w_r(i,12)).^2.*(((Delta_w-1).^2).^(1/(2.*...
            Coeff_w_r(i,12))-1)).^2+Coeff_w_r(i,1).*theta.*...
          (4/Coeff_w_r(i,12)).*((1/(2.*Coeff_w_r(i,12)))-1).*...
          ((Delta_w-1).^2).^(1/(2.*Coeff_w_r(i,12))-2));


        DD_DELTA_b_i = Coeff_w_r(i,2).*(Delta_w.^(Coeff_w_r(i,2)-1).*...
          DD_DELTA+(Coeff_w_r(i,2)-1).*Delta_w.^(Coeff_w_r(i,2)-
2).*D_DELTA.^2);

       D_psi = -2.*Coeff_w_r(i,9).*(Delta_w-1).*psi;
      DD_psi = (2.*Coeff_w_r(i,9).*(Delta_w-1).^2-1).*2.*Coeff_w_r(i,9).*psi;
       Comp = Coeff_w_r(i,6).*(DELTA.^Coeff_w_r(i,2).*(2.*D_psi+...

(Delta_w.*DD_psi))+2.*D_DELTA_b_i.*(psi+Delta_w.*D_psi)+DD_DELTA_b_i.*Delta_w
.*psi);

       Comp4 = Comp4+Comp;
       end

 DDa_w_r = Comp1+Comp2+Comp3+Comp4;

  %Ta_w_r

 % Comp1
  Comp1 = 0;
       for i = 1:7
        Comp = Coeff_w_r(i,6).*Coeff_w_r(i,5).*...
            Delta_w.^Coeff_w_r(i,4).*Tau_w.^(Coeff_w_r(i,5)-1);
        Comp1=Comp1+Comp;
       end
  % Comp 2

  Comp2 = 0;
       for i = 8:51
        Comp = Coeff_w_r(i,6).*Coeff_w_r(i,5).*Delta_w.^Coeff_w_r(i,4).*...
        Tau_w.^(Coeff_w_r(i,5)-1).*exp(-Delta_w.^Coeff_w_r(i,3));

        Comp2=Comp2+Comp;
       end

  % Comp3
  Comp3 = 0;
```

```
        for i = 52:54
         Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
         Tau_w.^Coeff_w_r(i,5).*exp(-Coeff_w_r(i,11).*...
           (Delta_w-Coeff_w_r(i,14)).^2-Coeff_w_r(i,12).*...
           (Tau_w-Coeff_w_r(i,13)).^2).*(Coeff_w_r(i,5)/Tau_w-...
           2.*Coeff_w_r(i,12).*(Tau_w-Coeff_w_r(i,13)));

         Comp3=Comp3+Comp;
        end


  % Comp4
  Comp4 = 0;
        for i = 55:56
        psi = exp(-Coeff_w_r(i,9).*((Delta_w-1).^2)-...
                  Coeff_w_r(i,10).*(Tau_w-1).^2);
         DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                  Coeff_w_r(i,1));
         T_DELTA_b_i = -2.*theta.*Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1);
         T_psi = -2.*Coeff_w_r(i,10).*(Tau_w-1).*psi;

         Comp =
Coeff_w_r(i,6).*Delta_w.*(T_DELTA_b_i.*psi+DELTA.^Coeff_w_r(i,2).*T_psi);

         Comp4=Comp4+Comp;
        end

  Ta_w_r = Comp1+Comp2+Comp3+Comp4;

  %TTa_w_r

 % Comp1
  Comp1 = 0;
        for i = 1:7
         Comp = Coeff_w_r(i,6).*Coeff_w_r(i,5).*(Coeff_w_r(i,5)-1).*...
             Delta_w.^Coeff_w_r(i,4).*Tau_w.^(Coeff_w_r(i,5)-2);
         Comp1=Comp1+Comp;
        end
  % Comp 2

  Comp2 = 0;
        for i = 8:51
         Comp = Coeff_w_r(i,6).*Coeff_w_r(i,5).*(Coeff_w_r(i,5)-1).*...
         Delta_w.^Coeff_w_r(i,4).*Tau_w.^(Coeff_w_r(i,5)-2).*...
         exp(-Delta_w.^Coeff_w_r(i,3));

         Comp2=Comp2+Comp;
        end

  % Comp3
  Comp3 = 0;
        for i = 52:54
         Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
         Tau_w.^Coeff_w_r(i,5).*exp(-Coeff_w_r(i,11).*...
           (Delta_w-Coeff_w_r(i,14)).^2-Coeff_w_r(i,12).*...
```

```
          (Tau_w-Coeff_w_r(i,13)).^2).*((Coeff_w_r(i,5)/Tau_w-...
          2.*Coeff_w_r(i,12).*(Tau_w-Coeff_w_r(i,13))).^2-...
          Coeff_w_r(i,5)/(Tau_w.^2)-2.*Coeff_w_r(i,12));

        Comp3=Comp3+Comp;
       end


  % Comp4
  Comp4 = 0;
       for i = 55:56
        DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                Coeff_w_r(i,1));
        T_DELTA_b_i = -2.*theta.*Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1);
        T_psi = -2.*Coeff_w_r(i,10).*(Tau_w-1).*psi;
        TT_DELTA_b_i = 2.*Coeff_w_r(i,2).*Delta_w.^(Coeff_w_r(i,2)-1)+...
        4.*theta.^2.*Coeff_w_r(i,2).*(Coeff_w_r(i,2)-1).*DELTA.^...
        (Coeff_w_r(i,2)-2);
        TT_psi = (2.*Coeff_w_r(i,10).*((Tau_w-1).^2)-
1).*2.*Coeff_w_r(i,10).*psi;

        Comp =
Coeff_w_r(i,6).*Delta_w.*(TT_DELTA_b_i.*psi+2.*T_DELTA_b_i.*T_psi+DELTA.^Coef
f_w_r(i,2).*TT_psi);

        Comp4=Comp4+Comp;
       end

  TTa_w_r = Comp1+Comp2+Comp3+Comp4;


  %DTa_w_r

 % Comp1
  Comp1 = 0;
       for i = 1:7
        Comp = Coeff_w_r(i,6).*Coeff_w_r(i,4).*Coeff_w_r(i,5).*...
            Delta_w.^(Coeff_w_r(i,4)-1).*Tau_w.^(Coeff_w_r(i,5)-1);
        Comp1=Comp1+Comp;
       end
  % Comp 2

  Comp2 = 0;
       for i = 8:51
        Comp = Coeff_w_r(i,6).*Coeff_w_r(i,5).*...
         Delta_w.^(Coeff_w_r(i,4)-1).*Tau_w.^(Coeff_w_r(i,5)-1).*...
        (Coeff_w_r(i,4)-Coeff_w_r(i,3).*Delta_w.^Coeff_w_r(i,3)).*...
        exp(-Delta_w.^Coeff_w_r(i,3));

        Comp2=Comp2+Comp;
       end

  % Comp3
  Comp3 = 0;
```

```
        for i = 52:54
         Comp = Coeff_w_r(i,6).*Delta_w.^Coeff_w_r(i,4).*...
         Tau_w.^Coeff_w_r(i,5).*exp(-Coeff_w_r(i,11).*...
           (Delta_w-Coeff_w_r(i,14)).^2-Coeff_w_r(i,12).*...
           (Tau_w-Coeff_w_r(i,13)).^2).*(Coeff_w_r(i,4)/...
           Delta_w-2.*Coeff_w_r(i,11).*...
           (Delta_w-Coeff_w_r(i,14))).*(Coeff_w_r(i,5)/Tau_w-...
           2.*Coeff_w_r(i,12).*(Tau_w-Coeff_w_r(i,13)));

         Comp3=Comp3+Comp;
        end


  % Comp4
  Comp4 = 0;
        for i = 55:56
         DELTA = theta.^2+Coeff_w_r(i,8).*(((Delta_w-1).^2).^...
                 Coeff_w_r(i,1));
         T_DELTA_b_i = -2.*theta.*Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1);
         T_psi = -2.*Coeff_w_r(i,10).*(Tau_w-1).*psi;
         TT_DELTA_b_i = 2.*Coeff_w_r(i,2).*Delta_w.^(Coeff_w_r(i,2)-1)+...
             4.*theta.^2.*Coeff_w_r(i,2).*(Coeff_w_r(i,2)-1).*DELTA.^...
             (Coeff_w_r(i,2)-2);
         TT_psi = (2.*Coeff_w_r(i,10).*((Tau_w-1).^2)-
1).*2.*Coeff_w_r(i,10).*psi;
         DT_psi = 4.*Coeff_w_r(i,9).*Coeff_w_r(i,10).*(Delta_w-1).*(Tau_w-
1).*psi;
         D_DELTA = (Delta_w-1).*(Coeff_w_r(i,7).*theta.*2/...
             Coeff_w_r(i,12).*((Delta_w-1).^2).^...
             (1/(2.*Coeff_w_r(i,12))-1)+2.*Coeff_w_r(i,8).*...
             Coeff_w_r(i,1).*((Delta_w-1).^2).^(Coeff_w_r(i,1)-1));
         DT_DELTA_b_i = -Coeff_w_r(i,7).*Coeff_w_r(i,2).*...
             (2/Coeff_w_r(i,12)).*DELTA.^(Coeff_w_r(i,2)-1).*(Delta_w-1).*...
             ((Delta_w-1).^2).^(1/(2.*Coeff_w_r(i,12))-1)-2.*theta.*...
             Coeff_w_r(i,2).*(Coeff_w_r(i,2)-1).*DELTA.^...
             (Coeff_w_r(i,2)-2).*D_DELTA;
         D_DELTA_b_i = Coeff_w_r(i,2).*DELTA.^(Coeff_w_r(i,2)-1).*D_DELTA;


         Comp = Coeff_w_r(i,6).*(DELTA.^Coeff_w_r(i,2).*(T_psi+...
             Delta_w.*DT_psi)+Delta_w.*D_DELTA_b_i.*T_psi+T_DELTA_b_i.*...
             (psi+Delta_w.*D_psi)+DT_DELTA_b_i.*Delta_w.*psi);


         Comp4=Comp4+Comp;
        end

  DTa_w_r = Comp1+Comp2+Comp3+Comp4;



%% Thermodynamic properties of Water
    % Pressure Calculations
     P_tild_w = R_w_tild.*T_w.*Rho_tild_w.*(1+Delta_w.*Da_w_r);
     P_w = P_tild_w;
```

```
            % Specific value of the property is calculated from Molar value
            %  as follow: x_tild_w = x_w/M_w
            %ie. Rho_tild_w = Rho_w/M_w

    % Isobaric Heat capacity
c_p_tild_w = R_w_tild.*(-(Tau_w.^2).*(TTa_w_0+TTa_w_r)+...
    ((1+Delta_w.*Da_w_r-Delta_w.*Tau_w.*DTa_w_r).^2)/...
    (1+2.*Delta_w.*Da_w_r+(Delta_w.^2).*DDa_w_r));
c_p_w = c_p_tild_w.*M_w;
    % Isochoric Heat Capacity
c_v_tild_w = -R_w_tild.*((Tau_w.^2).*(TTa_w_0+TTa_w_r));
c_v_w = c_v_tild_w.*M_w;
    % Enthalpy
h_tild_w = R_w_tild.*T_w.*(Tau_w.*(Ta_w_0+Ta_w_r)+Delta_w.*Da_w_r+1);
h_w = h_tild_w.*M_w;
    % Internal Energy
u_tild_w = R_w_tild.*T_w.*Tau_w.*(Ta_w_0+Ta_w_r);
u_w = u_tild_w.*M_w;
    % Entropy
s_tild_w = R_w_tild.*(Tau_w.*(Ta_w_0+Ta_w_r)-a_w_0-a_w_r);
s_w = s_tild_w.*M_w;
```