VIRTUAL MODELING AND OPTIMIZATION OF AN ORGANIC RANKINE CYCLE

THESIS

Presented in Partial Fulfillment of the Requirements for the Degree Master of Science in the Graduate School of The Ohio State University

By
Vetrivel Chandrasekaran, B.E.
Graduate Program in Mechanical Engineering

The Ohio State University
2014

Master’s Committee:
Prof. Marcello Canova, PhD, Advisor
Prof. Shawn Midlam-Mohler, PhD
ABSTRACT

Organic Rankine Cycles are used for Waste Heat Recovery from low temperature heat sources. In an Internal Combustion Engine, roughly one-third of the fuel energy is sent out through the exhaust. ORC’s were investigated for fuel efficiency improvements for heavy duty trucks in the 70s during the oil crisis. ORCs have once again gained interest with the current energy scenario and advances in technology. A recuperated ORC with R245fa as refrigerant is modeled in this thesis using the commercial 1-D simulation software GT-SUITE. The ORC is designed to extract energy from the exhaust of a gasoline powered light duty vehicle. Control inputs for the ORC are pump speed, expander speed and exhaust gas bypass valve position. The exhaust gas is not a steady source of heat, with varying temperature and mass flow rate depending on the operation of the vehicle. To maintain the ORC at a pre-determined operating state, feed-forward maps will be created. Exergy destruction is proposed to be used as a parameter that limits the control effort within reasonable limits. A second law analysis will be performed to identify points of exergy destruction and the ORC will be optimized using a Multi-Objective Particle Swarm Optimization algorithm to generate a Pareto front of net power output and exergy destruction in the system for a single exhaust condition that will allow the decision maker to choose a suitable state for the ORC. The Pareto front will be constructed for other off-design exhaust conditions and the trends will be observed between multiple exhaust gas conditions.
This page is intentionally wasted.
I wish to thank Dr. Canova, my advisor for giving me an opportunity to work on this project. Marco Cialesi Esposito for helping out with heat transfer concepts in the beginning of this project, Eric Lott for helping out with the exergy analysis and Junqiang Zhou for assisting in the controls.

I also wish to thank Chrysler Group LLC for sponsoring this project, especially Kevin Laboe and Mitchell Zajac. Chrysler provided us with component specifications and data that made modelling the ORC system possible.

Finally, I would like to thank Dr. T.C. Scott from University of Virginia, whose experience in thermo-fluid system was crucial in starting the research presented in this Thesis.
VITA

1990 ................................. Born in Salem, Tamil Nadu, India

2008-2012 .......................... B.E. in Automobile Engineering,

2012-2014 .......................... Graduate Research Associate,
                          The Ohio State University

FIELDS OF STUDY

Major Field: Mechanical Engineering

Specialization: Automotive Engineering
# TABLE OF CONTENTS

Abstract .................................................................................................................. ii  
Dedication ................................................................................................................. iii 
Acknowledgments ...................................................................................................... iv  
Vita ............................................................................................................................... v  
List of Figures ........................................................................................................... ix  
List of Tables ........................................................................................................... xiii  

CHAPTER PAGE

1 Introduction ............................................................................................................ 1  
1.1 Current Energy Scenario .................................................................................. 1  
1.2 Energy Efficiency Improvement Techniques ............................................... 4  
1.3 Waste Heat Recovery: Overview of Engineering Solutions ......................... 8  
  1.3.1 Turbocharging ............................................................................................... 9  
  1.3.2 Turbocompounding ...................................................................................... 11  
  1.3.3 Thermoelectric Generators ........................................................................ 13  
1.4 Organic Rankine Cycles .................................................................................... 15  
  1.4.1 Operation of ORC ........................................................................................ 16  
  1.4.2 Fluid Selection ............................................................................................. 18  
1.5 Usage in the Automotive Industry .................................................................... 20  
  1.5.1 Difficulties in Vehicle Implementation ....................................................... 21  
  1.5.2 ORC Control ............................................................................................... 23  
1.6 Objectives of the Thesis .................................................................................... 24  

2 Modeling of the Organic Rankine Cycle System .............................................. 26  
2.1 Layout of System ............................................................................................. 26  
2.2 Introduction to GT-SUITE ............................................................................... 32  
2.3 Heat Exchanger Models Setup and Calibration .............................................. 35  
  2.3.1 Geometric Data ............................................................................................ 37
2.3.2 Heat Transfer Models ........................................... 38
2.3.3 Pressure Drop Data ............................................. 44
2.3.4 Calibration of the Evaporator Model ....................... 48
2.3.5 Calibration of the Recuperator Model ...................... 50
2.3.6 Calibration of the Condenser Model ......................... 52
2.4 Pump Model Setup ............................................... 53
2.5 Expander Model Setup ............................................ 54
2.6 Receiver Model Setup ............................................ 54
2.7 Assembly & Verification ........................................ 54
   2.7.1 Initialization Interface and Parameters ................ 55
   2.7.2 Solver Settings ............................................. 56
   2.7.3 Model Tuning ............................................... 58
2.8 System Inputs/Outputs and Constraints ....................... 62
2.9 The Design Point ............................................... 64
2.10 Operation of ORC system at Off-Design Conditions ........ 69
2.11 Feed-Forward Control .......................................... 73

3 Introduction to Particle Swarm Optimization ..................... 77
   3.1 Classification of Optimization Problems ................... 77
   3.2 Solutions to Optimization Problems ......................... 79
      3.2.1 Overview of Iterative Optimization Methods for Static Non-Linear Problems ......................... 80
      3.2.2 Metaheuristic Algorithms ................................ 83
   3.3 Particle Swarm Optimization ................................ 83
      3.3.1 Algorithm Description ................................... 84
         3.3.1.1 Constraint Handling ................................ 90
      3.3.2 Description of Sample Problem ......................... 94
      3.3.3 Application of PSO to Sample Problem, Results and Comparison with Other Optimization Methods .... 95
      3.3.4 Implementation of PSO for Multiobjective Optimization .. 97
         3.3.4.1 MOPSO algorithm .................................... 101
      3.3.5 Example Problem ......................................... 104

4 Analysis and Optimization of the ORC .......................... 107
   4.1 Design of Experiments and Analysis ......................... 107
      4.1.1 Second Law Analysis .................................... 113
      4.1.2 Exergy Destruction in the ORC System ................. 115
      4.1.3 The Optimization Problem ............................... 124
   4.2 Multi-Objective Optimization with PSO ...................... 125
   4.3 Setup .......................................................... 126
   4.4 Results .......................................................... 130
      4.4.1 Parameterization of the Pareto Front ................ 130
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.4.2</td>
<td>Pareto Analysis</td>
<td>134</td>
</tr>
<tr>
<td>4.4.3</td>
<td>Selection of Operating Point</td>
<td>137</td>
</tr>
<tr>
<td>4.4.4</td>
<td>Cycle Analysis</td>
<td>139</td>
</tr>
<tr>
<td>4.4.5</td>
<td>Evaporator analysis</td>
<td>141</td>
</tr>
<tr>
<td>4.5</td>
<td>Pareto Front for Operation of the ORC at Off-Design conditions</td>
<td>145</td>
</tr>
<tr>
<td>4.6</td>
<td>Conclusion</td>
<td>150</td>
</tr>
<tr>
<td>4.7</td>
<td>Future Work</td>
<td>151</td>
</tr>
</tbody>
</table>

Bibliography ................................................................. 152
# LIST OF FIGURES

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Oil consumption by region (million bpd) [1]</td>
<td>2</td>
</tr>
<tr>
<td>1.2 Trend of crude oil prices [2]</td>
<td>3</td>
</tr>
<tr>
<td>1.3 Future fuel consumption mandates [3]</td>
<td>3</td>
</tr>
<tr>
<td>1.5 Vehicle onboard electrical consumption [4]</td>
<td>5</td>
</tr>
<tr>
<td>1.6 Layout of a turbocharger [5]</td>
<td>9</td>
</tr>
<tr>
<td>1.7 Layout of a turbocompounding system [6]</td>
<td>11</td>
</tr>
<tr>
<td>1.8 Simulated BSFC comparison between m-TC and e-TC [7]</td>
<td>13</td>
</tr>
<tr>
<td>1.9 Temperature-ZT relation [8]</td>
<td>14</td>
</tr>
<tr>
<td>1.10 ZT development over the years [8]</td>
<td>14</td>
</tr>
<tr>
<td>1.11 ORC with Recuperator</td>
<td>16</td>
</tr>
<tr>
<td>1.12 T-s and p-H diagram for an ORC cycle</td>
<td>18</td>
</tr>
<tr>
<td>1.13 Fluid classification based on saturation curve</td>
<td>18</td>
</tr>
<tr>
<td>2.1 ORC plant diagram</td>
<td>26</td>
</tr>
<tr>
<td>2.2 Plate heat exchanger [9]</td>
<td>28</td>
</tr>
<tr>
<td>2.3 Coolant pump performance maps</td>
<td>31</td>
</tr>
<tr>
<td>2.4 Staggered grid [10]</td>
<td>34</td>
</tr>
<tr>
<td>2.5 Evaporator standalone model</td>
<td>36</td>
</tr>
<tr>
<td>2.6 Evaporator HxMaster dialog box</td>
<td>36</td>
</tr>
</tbody>
</table>
2.7 Evaporator HeatExchangerSpecs dialog box .......................... 37
2.8 HeatExchangerSpecs template options tree ........................... 37
2.9 Heat exchanger internal structure ..................................... 38
2.10 Predictive correlations for each phase of the refrigerant ........... 48
2.11 Evaporator calibration correlation plots .............................. 49
2.12 Recuperator calibration correlation plots ............................. 51
2.13 Condenser calibration correlation plots ............................... 52
2.14 ORC GT-SUITE model .................................................. 55
2.15 GT-SUITE case setup dialog box ..................................... 56
2.16 Example of boundary condition initialization ramps to improve convergence ......................................................... 60
2.17 Effect of total refrigerant mass in the system on subcooling at the condenser outlet ......................................................... 61
2.18 ORC plant diagram ....................................................... 65
2.19 T-s and p-H diagram for an ORC Cycle .............................. 66
2.20 Energy flow through the ORC at design point ....................... 68
2.21 Exhaust conditions at different vehicle speeds ...................... 69
2.22 System parameters vs vehicle speed ................................... 71
2.23 ORC feed-foward control schematic ................................. 73
2.24 ORC feed-forward maps ................................................ 74
2.25 ORC feed-forward map performance .................................. 75
3.1 Graphical illustration of movement of one particle within the search space ................................................................. 86
3.2 Violated design point redirection. (a) Particle outside feasible region. (b) Particle position redirection. [11] ....................... 91
3.3 Search space restriction method [12] .................................. 92
3.4 Rosenbrock function ..................................................... 95
3.5 Snapshots of the MOPSO at various iterations ....................... 105
<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1</td>
<td>Normalized $P_e$ vs $\eta_{ORC}$ results from DOE</td>
<td>109</td>
</tr>
<tr>
<td>4.2</td>
<td>Normalized $P_e$, $Q_{evap}$ and $\eta_{ORC}$ results from DOE with progression of $N_e$ detailed. Each line connects data points with identical $x_{BP}$ and $N_{fp}$ but different $N_e$.</td>
<td>111</td>
</tr>
<tr>
<td>4.3</td>
<td>Effect of $T_4$ and $p_H$ on a) cycle efficiency [-] (normalized w.r.t maximum efficiency in plot) b) Fraction of total heat supplied to the refrigerant by recuperator [-].</td>
<td>112</td>
</tr>
<tr>
<td>4.4</td>
<td>Component-wise exergy destruction plotted with respect to $N_p$, $N_e$ and $x_{BP}$.</td>
<td>118</td>
</tr>
<tr>
<td>4.5</td>
<td>Component’s contribution to total exergy destruction plotted with respect to $N_p$, $N_e$ and $x_{BP}$.</td>
<td>118</td>
</tr>
<tr>
<td>4.6</td>
<td>Evaporator exergy efficiency [-] as a function of $\Delta T_{in, evap}$ and $p_H$.</td>
<td>120</td>
</tr>
<tr>
<td>4.7</td>
<td>Exergy-Enthalpy plot for air [13].</td>
<td>121</td>
</tr>
<tr>
<td>4.8</td>
<td>Illustration of exergy loss through exhaust mixing.</td>
<td>122</td>
</tr>
<tr>
<td>4.9</td>
<td>(a) Expander exergy efficiency [-] as a function of $T_4$ and $p_H$ (b) Expander T-s diagram [14].</td>
<td>123</td>
</tr>
<tr>
<td>4.10</td>
<td>System exergy destruction vs net expander power.</td>
<td>124</td>
</tr>
<tr>
<td>4.11</td>
<td>Pareto optimal points generated from the MOPSO overlaid on DOE data points.</td>
<td>130</td>
</tr>
<tr>
<td>4.12</td>
<td>$F(x)$ with $\alpha$ sweep.</td>
<td>132</td>
</tr>
<tr>
<td>4.13</td>
<td>Pareto front at 65 mph with $\alpha$ overlaid.</td>
<td>133</td>
</tr>
<tr>
<td>4.14</td>
<td>Control input constraints of the pareto optimal points.</td>
<td>134</td>
</tr>
<tr>
<td>4.15</td>
<td>System constraints space plot.</td>
<td>135</td>
</tr>
<tr>
<td>4.16</td>
<td>Constraints of the Pareto optimal points.</td>
<td>136</td>
</tr>
<tr>
<td>4.17</td>
<td>Cycle analysis of ORC for various points on the Pareto front.</td>
<td>139</td>
</tr>
<tr>
<td>4.18</td>
<td>Evaporator profile for various points on the Pareto front.</td>
<td>141</td>
</tr>
<tr>
<td>4.19</td>
<td>Pareto optimal points generated from the MOPSO for 57 mph case.</td>
<td>145</td>
</tr>
<tr>
<td>4.20</td>
<td>Constraints for the 57 mph case. (a) Control input constraints (b) System constraints.</td>
<td>146</td>
</tr>
<tr>
<td>4.21</td>
<td>Constraints of the Pareto optimal points.</td>
<td>147</td>
</tr>
</tbody>
</table>
4.22 Design point and off-design Pareto front comparison normalized with respect to 65 mph extreme points
# LIST OF TABLES

<table>
<thead>
<tr>
<th>TABLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Contribution to fuel economy improvements [15]</td>
</tr>
<tr>
<td>2.1</td>
<td>Evaporator calibration error summary</td>
</tr>
<tr>
<td>2.2</td>
<td>Recuperator calibration error summary</td>
</tr>
<tr>
<td>2.3</td>
<td>Condenser calibration error summary</td>
</tr>
<tr>
<td>2.4</td>
<td>FlowControl defaults for WHR systems</td>
</tr>
<tr>
<td>2.5</td>
<td>Boundary variables for the ORC</td>
</tr>
<tr>
<td>2.6</td>
<td>ORC physical constraints</td>
</tr>
<tr>
<td>2.7</td>
<td>Characterization of Design Point</td>
</tr>
<tr>
<td>2.8</td>
<td>ORC energy balance</td>
</tr>
<tr>
<td>2.9</td>
<td>Exhaust energy breakup</td>
</tr>
<tr>
<td>2.10</td>
<td>Relevant Parameters of the Vehicle and ORC at Different Vehicle Operating Points</td>
</tr>
<tr>
<td>3.1</td>
<td>Step direction update equations [16]</td>
</tr>
<tr>
<td>3.2</td>
<td>PSO settings</td>
</tr>
<tr>
<td>3.3</td>
<td>Benchmarking results</td>
</tr>
<tr>
<td>3.4</td>
<td>MOPSO suggested range for tuning parameters [17]</td>
</tr>
<tr>
<td>3.5</td>
<td>PSO parameters used in example problem</td>
</tr>
<tr>
<td>4.1</td>
<td>Characterization of design point</td>
</tr>
<tr>
<td>4.2</td>
<td>ORC physical constraints</td>
</tr>
</tbody>
</table>
4.3 Exergy destruction equations for individual components of the ORC [18]116
4.4 Exergy data for the ORC at design point . . . . . . . . . . . . . . . . 117
4.5 Summary of PSO parameters for the Pareto optimization . . . . . . 129
4.6 Summary of parameters for Pareto optimal points at 65 mph . . . . 138
4.7 Evaporator parameters for each phase of the refrigerant . . . . . . 144
4.9 Design Point at 57mph . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 147
4.10 Summary of parameters for Pareto optimal points at 57 mph . . . . 149
CHAPTER 1
INTRODUCTION

Governments all over the world are imposing stricter vehicle fuel efficiency standards with the purpose of reducing dependency on foreign oil imports, lessen environmental impact due to burning of fossil fuels and also decouple the auto industry from its overwhelming reliance on fossil fuels. The U.S. government’s CAFE regulations enacted in response to the Middle East’s oil embargo in the 1970’s and updated regularly ever since, has forced auto makers to come up with ingenious solutions to reduce fuel consumption and emissions [19]. Renewed fear of running out of oil supplies which manifests in steadily increasing oil prices on the international market has also influenced the push towards reducing fuel consumption.

1.1 Current Energy Scenario

At the end of 2012, world oil reserves were estimated to last for 52.9 years [1]. Though new reserves are being continuously discovered (at decreasing frequency), the increase in energy consumption of developing countries, especially highly populated ones such as China and India will only increase the demand for oil in the future. Figure 1.1 illustrates the trend in oil consumption divided into regions. With China and India put together accounting for 15.9% of the total oil consumption and their demand for oil growing at around 5% every year, it is only a matter of time until either all oil
reserves are depleted or it become too expensive to extract thus making oil practically unviable to fuel our transportation system [1].

Figure 1.1: Oil consumption by region (million bpd) [1]

Figure 1.2 shows the trend in crude oil prices from the 1990’s [2]. Global oil prices have been steadily increasing from the beginning of the new millennium, breaching the $100 mark in ~2008 but sharply dropping soon afterwards due to the economic recession in the U.S and Europe. However, prices have recovered to the pre-recession levels with current oil prices looming at levels higher than during the Middle East Oil Embargo and the Iranian Revolution, even accounting for inflation [1, 20]. This indicates that the era of cheap oil is over, putting significant pressure on the auto industry to improve fuel economy.

To respond to the increasingly hostile economic situation and due to environmental concerns, nations all over the world have imposed ever tightening fuel efficiency
and CO$_2$ standards on production vehicles. Figure 1.3 summarizes fuel efficiency standards mandated by various countries. In the US, the gradual tightening of fuel economy and hence CO$_2$ emissions is mandated by the Corporate Average Fuel Economy (CAFE) standards [19]. To satisfy regulatory requirements and consumer demand for higher fuel efficiency, auto manufacturers have come up with various technology implementations to overcome these challenges.

![Figure 1.2: Trend of crude oil prices [2]](image)

![Figure 1.3: Future fuel consumption mandates [3]](image)
1.2 Energy Efficiency Improvement Techniques

An understanding of how the fuel energy is utilized in a vehicle would provide a clearer picture of where energy is wasted in the system thus allowing us to focus on reducing wastage at these points.

The internal combustion engine, which propels an overwhelming majority of vehicles on the road, is an inherently inefficient power conversion device. In the process of burning fuel to generate useful mechanical work, an IC engine wastes more than half of the fuel energy primarily through two paths, namely, the exhaust and cooling system. Other minor sources of inefficiency too exist in a vehicle, with Figure 1.4 providing a detailed break up of how fuel energy is used in a mid-sized sedan for the EPA city and highway cycles. It is immediately evident that the exhaust and cooling system are the source of one-third of the energy losses each, with braking energy being the next major source of losses. Braking energy encompasses transmission losses, tire rolling resistance, aerodynamic resistance, idling and coasting; and braking losses. Ancillary loads such as the coolant water pump, alternator, power steering pump, etc. contribute to the accessory loads.
Recently, there is a trend of increasing accessory loads spurred by higher electrical power requirements in an automobile due to multiple power hungry electronic devices being installed on-board the vehicle such as seat warmers, GPS navigation, Parking assistance cameras, infotainment screens, climate control, high end audio systems, etc. The yearly trend of average power requirements of the electronics on-board a vehicle is shown in Figure 1.5.
This increase in power consumption causes the alternator to put a greater load on the engine, with alternators absorbing 2-5% or in the range of 5 kW of an engine’s brake horsepower [21, 22]. Thus auto manufacturers face the problem of having to manage fuel efficiency improvements in the face of increasing engine loads.

Improved part load performance through engine downsizing has been the focus of automakers, with various approaches to achieve lower fuel consumption being implemented. Engine downsizing without sacrificing on performance can be achieved through use of turbocharging, Gasoline Direct Injection (GDI), variable valve actuation, cylinder deactivation, reducing component friction and hybridization to name a few.

The Vehicle Technology Program of the U.S. Department of Energy has co-sponsored projects in collaboration with a few companies from auto industry with the aim of reducing vehicle fuel consumption by 26% compared to a reference powertrain. In this program, Chrysler’s demonstrated a reduction of fuel consumption by upto 26% from a baseline 4L V6 engine with a 6-speed transmission through downsizing, reducing accessory losses, improved thermal management and transmission improvements [15]. Table 1.1 shows the contribution of each system to total fuel economy improvements.

Engine efficiency improvements were obtained through implementation of high cylinder compression ratio, cooled EGR, dual fuel mode (diesel+gasoline), advanced fuel injection techniques and low friction bore liners. The engine was downsized to a 2.4L I4 engine. A 9-speed gearbox helped maintain the engine near the ideal operating region. The torque converter was locked at low speeds to reduce transmission losses. Engine warm-up time was reduced by rerouting engine coolant heat to the engine and transmission oil reducing viscosity and hence friction. Also advanced optimization
<table>
<thead>
<tr>
<th>Type</th>
<th>% Contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engine efficiency</td>
<td>12</td>
</tr>
<tr>
<td>Ideal engine operation</td>
<td>6</td>
</tr>
<tr>
<td>Low lock-up speed</td>
<td>6</td>
</tr>
<tr>
<td>Thermal management</td>
<td>1</td>
</tr>
<tr>
<td>Reduced losses</td>
<td>1</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>26</strong></td>
</tr>
</tbody>
</table>

algorithms were implemented to reduce power consumption by the electric coolant water pump, radiator fan and alternator.

Ford’s approach in this program, is more focused on engine downsizing through improving the combustion process by implementing techniques such as lean combustion with gasoline direct injection and dilute combustion with cooled EGR, all of which is incorporated in their EcoBoost™ technology [23]. Better engine control, low friction components and NVH improvements were also implemented. The engine is downsized from a 3.5L V6 to a turbocharged 2.3L in-line 4 cylinder engine. Ford was able to demonstrate a total fuel economy improvement of 28%.

If the reader is interested, [24] and [25] provide a review of the technologies that can be implemented to improve fuel economy and achieve engine downsizing in the short to medium term.

From the above examples, it is clear that there still is scope for improvements in engine efficiency through advanced combustion techniques but these improvements are soon approaching the limits of cost-effectiveness thus requiring the need for other
avenues of engine efficiency improvements. The exhaust seems to be good candidate to achieve powertrain efficiency improvements in a cost-effective method.

1.3 Waste Heat Recovery: Overview of Engineering Solutions

With improvements to the combustion process providing diminishing returns, exhaust waste heat recovery is turning to be a very attractive route to achieving fuel economy improvements due to its high exergy content since the exhaust gas temperature of a gasoline engine is in the range of 500-900°C [26, 27]. Multiple techniques have been developed over the years to harness this rich source of exergy into useful work. An outline of current and emerging technologies implemented to harvest energy from the exhaust will be provided. There are 4 mechanisms that have been in various stages of implementation in recovering waste heat from the exhaust gases. They are:

1. Turbocharging
2. Turbocompounding
3. Thermoelectrics
4. Organic Rankine cycles

Of these, turbocharging is the most widely used and well established method of waste heat recovery, having been implemented in almost all classes of vehicles ranging from sub-compacts to heavy duty commercial vehicles.
1.3.1 Turbocharging

Exhaust gas turbocharging consists of utilizing a turbine to produce mechanical power from expanding the hot gases in the exhaust manifold. This power is then used to drive a compressor at the intake manifold. The turbine and compressor are usually centrifugal. The kinetic energy of the hot exhaust gas spins the turbine which in turn drives the compressor. The compressor forces air into the combustion chamber increasing the power output of the engine by virtue of more air trapped in the cylinder and since the turbocharger uses energy otherwise wasted into the atmosphere for the compression process, the overall efficiency of the engine increases [28–30]. There might be a small energy penalty imposed by increased back pressure at the exhaust caused by the turbine installation restricting exhaust flow [28]. The intake air compression process also increases charge temperature. Charge Air Coolers (CAC) are usually employed right after the compressor to lower the temperature of the charge before it enters the combustion chamber [31]. The consequent increase in density of air also improves volumetric efficiency.

In turbocharged SI engines, since more air is trapped in the cylinder, more fuel has to be sent into the chamber to maintain stoichiometric air-fuel ratio. Therefore a turbocharged engine with the same displacement as a naturally aspirated engine would
have higher fuel consumption but with better power and torque characteristics. The higher air density in the combustion chamber also increases pressure and temperature during the compression stroke raising the engine’s susceptibility to knocking. This forces the reduction of compression ratio and/or requirement for fuels with higher octane rating [31]. The rise in charge temperature during compression diminishes knock resistance which to an extent can be mitigated by a CAC. Other countermeasures to reduce knock involve running the engine with a richer air-fuel mixture to further cool the charge and employing knock sensors to automatically adjust spark timing [31].

Turbocharging compression-ignition (CI) engines has less of a problem with engine knock since the fuel is injected just before combustion. The fact that CI engines are not throttle controlled also makes implementation simpler. Major factors that limit boost pressures and high compression ratio’s in CI engines are mechanical and thermal loading of engine components [28, 31]. The relative ease with which turbochargers can be implemented has contributed to their prevalence in CI engines.

Turbocharging plays a significant role in engine downsizing due to the ability to increase power densities and offer better part load performance, leading to overall fuel economy improvements through lower pumping losses, smaller heat transfer surfaces, less friction and a lighter overall engine packaging [24,31,32]. By reducing the number of cylinder, piston friction can be avoided and also downsized engines allow for smaller bearing diameters [24]. Turbochargers can also be used in combination with other WHR mechanisms with the turbine usually being placed upstream [6].

With turbocharging playing a major role in engine downsizing and coupled with gasoline direct injection technology, significant fuel economy improvements have been reported [24,29,33].
1.3.2 Turbocompounding

Figure 1.7: Layout of a turbocompounding system [6]

The principle of turbocompounding is similar to turbocharging, however in this case the turbine is mechanically coupled to the engine crankshaft through a gearbox and hydraulic coupling, instead of a compressor [7].

Turbocompounding was first used in rotary aircraft engines after the end of World War II when engineers realized that there was a huge potential to recover energy from the exhaust gases [34, 35]. However, the technology became rapidly obsolete due to the advent of gas turbine engines [7]. Turbocompounding is nowadays primarily used in heavy duty trucks, where it is considered a cheap and reliable solution for exhaust heat recovery [36]. Figure 1.7 depicts a typical schematic of an truck engine with turbocompounding and a turbocharger [6]. The turbocompounding device is usually placed downstream of the turbocharger for better dynamic response during engine load transients [6].

In 1981, Cummins developed a turbocompound engine claiming a 4.6% improvement in fuel consumption on a 14L 6-cylinder engine. The turbine was connected to
the crankshaft by means of a gear reduction and fluid coupling [35]. In 1986, Caterpillar experimented with turbocompound engines, claiming a 8.5% improvement in fuel consumption in their 11.3L 6 cylinder engine. The turbine powered the engine coolant pump and was also connected to the crankshaft [35]. Scania introduced their first engine incorporating a mechanical turbocompound system, in 1991. The turbine was coupled to the crankshaft by means of a hydrokinetic clutch and toothed gear, claiming fuel savings of 5% [7,35]. Daimler introduced a turbocompound engine that was claimed to be 5% more efficient than their previous engine line [7]. Similar to turbocharging one major drawback with turbocompounding is the effect of backpressure. Though the turbine produces a net increase in power at high loads, at low load conditions it acts as a restriction resulting in a net loss of power but with a turbocharger present upstream energy recovery at low loads is possible [6,30]. Decoupling the mechanical connection between the compound turbine and crankshaft and connecting it to a generator instead, serves to prevent it from sapping power from the engine at low load conditions.

The current research in turbocompounding is exploring the option of generating electrical power rather than mechanically connecting the turbine to the crankshaft. In electrical turbocompounding the turbine is not constrained to rotate at the speed of crankshaft, hence its speed can be controlled to optimize efficiency while also allowing more control over the power generation [37]. The electrical energy generated can be stored in a battery or an ultracapacitor that can be used to power instruments on-board or provide extra power to the crankshaft through a traction motor [37]. Simulation results (assuming 85% overall efficiency from the turbine generator to the crankshaft motor) provided a 5-10% fuel consumption improvement on a C-15 Caterpillar medium duty on-highway truck engine [37].
Figure 1.8 shows simulation data of fuel consumption comparison between electrical and mechanical turbocompounding on a 10L heavy duty engine for a variety of driving cycles. It can be seen that electrical turbocompounding has consistently lower BSFC on all the driving cycles tested.

![Fuel Consumption Comparisons](image)

Figure 1.8: Simulated BSFC comparison between m-TC and e-TC [7]

### 1.3.3 Thermoelectric Generators

Thermoelectric generators (TEG) are direct energy conversion devices that create a voltage in the presence of a temperature gradient between two sides of the TEG. The operation of a TEG is based on three thermoelectric effects, namely, the Seebeck effect, the Peltier effect and the Thompson effect [21]. The performance of a thermoelectric material is evaluated on the basis of its *figure of merit* (ZT),

\[ ZT = \frac{\alpha^2 \lambda}{\rho T} \]

where

- \( \alpha \) Seebeck coefficient
- \( \lambda \) electrical conductivity
- \( \rho \) thermal conductivity
- \( T \) temperature
ZT is highly sensitive to the temperature of the heat source and a suitable TE material is usually selected based on this parameter. Figure 1.9 shows the temperature dependence of ZT for various materials [8].

![Figure 1.9: Temperature-ZT relation](image1)

TEG have potential to replace alternators [8,21,22,38] or be used alongside them, reducing load on the engine and improving fuel consumption.

A rich literature is available on TEG’s based on Bi$_2$Te$_3$ [21, 38, 39], SiGe [40], FeSi$_2$ [8, 41], PbTe [8] and skutterudites [42] being implemented in vehicles, however none have been successful enough to be commercially adopted. The major hindrances include low ZT of the material, thus producing insufficient electricity, commercial non-availability and cost of materials [21, 22, 26, 43]. One major issue with the commercially available TE material Bi$_2$Te$_3$ is its low ZT value and ZT-temperature profile being unsuitable for use in exhaust gas heat recovery. Bi$_2$Te$_3$ investigations have been done using additional mechanisms like heat pipes [44] to reduce the exhaust gas temperatures flowing over the thermoelectric materials, however the lower temperature difference limits the power generated from the TEG and therefore its efficiency. To work within the limits imposed by the TEG material, research has been done to extract power from the engine cooling system [22], a low temperature heat source.
Recently there has been a push towards finding new materials with a higher \( ZT \) (see Figure 1.10) [8]. New materials such as quantum dots, quantum wells, nanowires and other materials suitable for the high temperature of the exhaust are still under research [45,46].

### 1.4 Organic Rankine Cycles

The Organic Rankine Cycle (ORC) is currently a promising solution for waste heat recovery in IC engines, which can be applied to extract energy from the engine exhaust system (or the cooling circuit). The ORC is a vapor powered thermodynamic cycle that employs a high molecular weight organic refrigerant as its working fluid. ORC’s are suitable for recovering low-grade heat that is otherwise wasted. ORC’s have been used in small scale power generation plants that use solar, bio-mass or geothermal energy; and also for waste heat recovery from energy intensive industrial processes due to their cost-effectiveness.

A steam based Rankine cycle cannot be used to recover energy from low temperature heat sources due to the thermodynamic properties of water, hence the entry of ORC’s for this purpose. The development of ORC technology has helped recover low temperature heat previously considered cost-prohibitive. Low maintenance and equipment costs of ORC’s also serve as another advantage [47].

Literature is available on experimental studies of using ORC’s in solar power plants [48], numerous cost-benefit analyses [49–52] and finally actual solar ORC plants in commercial operation [47,53,54]. Literature studies also exist on their implementation to convert geothermal energy [55–57]. Geothermal power plants based on the ORC have been constructed in California, Hawaii, Philippines, New Zealand, Australia and many other locations all over the world [54,58].

ORC’s have also been used as bottoming cycles in various industries where a huge
amount of heat is rejected as waste. There are numerous studies of ORC’s for this purpose [59–63], with actual implementation in the cement industry [64, 65], Wood processing industry [66] and furniture production industry [67].

Before going into their application in the automotive industry, the working of the ORC will be briefly touched upon in the next section.

1.4.1 Operation of ORC

The basic configuration of the Organic Rankine Cycle requires four components, namely, an evaporator, expander, condenser and feed pump. The addition of a recuperator helps increasing cycle efficiency. The components of a regenerative ORC are arranged as shown below:

![Figure 1.11: ORC with Recuperator](image_url)
The thermodynamic process is summarized as follows:

1-2: The subcooled working fluid is pressurized by the pump to the evaporator pressure.

2-3: The pressurized fluid is preheated at constant pressure in the recuperator.

3-4: An isobaric heat addition from the hot exhaust gas causes a fluid phase change to vapor.

4-5: Energy is extracted from the working fluid by expanding it down to the condenser pressure.

5-6: The expander working fluid exchanges heat with the pressurized liquid.

6-1: An Isobaric heat rejection to a coolant causes phase change to liquid.

The thermal efficiency of the cycle is given as:

\[ \eta_{TH} = \frac{P_e - P_{fp}}{\dot{Q}_{evap}} = \frac{P_{net}}{\dot{Q}_{evap}} \]  

(1.4.1)

where

- \( P_e \) expander power output
- \( P_{fp} \) feed pump power
- \( \dot{Q}_{evap} \) heat input through the evaporator

Practical issues such as pressure drop in the heat exchangers, irreversibilities in the expander, heat loss and friction in piping will typically reduce \( \eta_{TH} \) to 8-15%. The T-s and p-H diagram for the cycle using R-245fa (a dry fluid) is shown in Figure 1.12. Note that the expander exhaust is still superheated. Rather than rejecting all this heat to the condenser, the recuperator recycles this energy back into the system by pre-heating the pressurized fluid going into the evaporator.
1.4.2 Fluid Selection

Fluids can be classified into three types based on the slope of the vapor saturation line on the T-s diagram (Figure 1.13). Fluids with a positive slope, negative slope and straight line of vapor saturation line are called dry, wet and isentropic fluids respectively.

An advantage of dry fluids is that the fluid need not be superheated before entering the expander. Fluid droplets can damage the expander blades in a turbine expander or damage the seals in the case of a scroll expander. Wet fluids have to be superheated...
before being sent to the expander otherwise the fluid might enter the two-phase region in the middle of the expansion process. The absence of superheating in dry fluids allows for usage of more compact heat exchangers as vapor has inferior heat transfer characteristics, requiring a larger heat transfer area [48]. Also, cycle efficiency of non-recuperative ORC’s is weakly dependent on evaporator superheat and is detrimental to cycle efficiency with dry fluids [68, 69].

Working fluid selection is based on factors such as temperature of heat source, safety, environmental impact, fluid cost, chemical stability and other economical and physical constraints (e.g. fluid mass flow rate limits, system packaging constraints etc.). There does not exist a single fluid that can be considered optimum for use in all ORC’s when considering all the aforementioned factors.

Good working fluids should have properties such as low toxicity and low potential for flammability (for reasons of safety), chemical stability and high thermal stability. The fluids must also be environmentally safe by having a low Ozone Depletion Potential (ODP) and Global Warming Potential (GWP) which makes for easier handling. The working fluid should also be non-corrosive to all materials that it comes into contact with during operation.

High density fluids provide high power output/component weight ratio allowing smaller, lighter and cheaper equipment [69–71]. Fluids with high boiling points will have higher efficiencies because of a larger area occupied by the cycle on the T-s diagram [68, 72]. [73] reported that high latent heat at low pressures results in a significant amount of heat being rejected at the condenser, lowering overall efficiency. At the boiler, high latent heat helps in better heat absorption due to superior heat transfer characteristics in two-phase [74]. [75, 76] claims that lower latent heat increases fluid mass flow rate increasing expander power output.
1.5 Usage in the Automotive Industry

In the automotive industry, the usage of ORC’s for waste heat recovery was first investigated in trucks during the 1970’s oil crisis. Implementing an ORC waste heat recovery system in trucks is easier due to their predominant highway usage providing a steady flow of exhaust heat with relatively less transients when compared to passenger vehicles. [77] is one of the first publications to demonstrate usage of ORC’s in vehicles. A 288 HP Mack 676 diesel engine fitted with an ORC showed a 12.5% improvement in fuel economy over a 450 km on-road test. [78] discusses the installation of an ORC WHR system in a Mack Trucks diesel engine and components used along with plans for a 1-year test program. Interest in automotive ORC WHR systems waned after the 1970’s oil crisis was resolved until a resurgence in interest recently due factors such as steady uptick in oil prices, improved technology and ever tightening fuel economy and emissions standards.

Cummins is in the process of incorporating an ORC in their line of Class 8 trucks with a goal of 10% fuel efficiency with their test ORC system using heat from both the EGR (working fluid acting as an EGR cooler) and exhaust to heat the working fluid [79].

One of the earliest investigations of ORC’s in passenger vehicles was done by Toyota in 1993 [80]. A simple ORC using R-123 as the working fluid was integrated with an engine on a test bench. The water jacket of the engine block was used as the evaporator and coolant radiator was re-purposed as a condenser. The energy extracted under three conditions: hill-climbing, 40 kmph cruising and idling were 400 W, 240 W and 160 W respectively. More recently, Honda [81] and BMW [82,83] have investigated using Rankine cycles in their passenger vehicles. Honda’s implementation used a steam Rankine cycle in their hybrid vehicle [81]. A swash plate axial piston type expander was connected to an electrical generator. The electrical energy generated was
used to drive a traction motor that was brought online during vehicle acceleration. Rankine cycle efficiency was reported to be 13.8%. BMW’s ”Turbosteamer” concept uses a dual loop steam Rankine cycle with an estimated 5-10% fuel savings [84].

1.5.1 Difficulties in Vehicle Implementation

While designing an ORC system for exhaust heat recovery one has to take in account considerations such as system efficiency, weight, packaging constraints, component costs, working fluid properties and effect of the WHR system on other components of the vehicle. Many of these considerations are intertwined with each other.

The ultimate aim of the ORC would be to obtain high cycle efficiency and power output. This can be achieved through higher system pressure ratio’s [85–89] but it also increases complexity of system design by requiring better fittings, higher metal thickness for heat exchangers increasing the weight and cost of the entire system [27]. Higher pressure ratio’s also increase the complexity of expander design entailing use of multi-stage expanders, again increasing the cost and size of the expander [27], however the swept volume required may be reduced due to increase in vapor density [70]. The cost of the feed pump will also appreciate due to higher pressure differential [27,70,86].

Factors that affect expander selection are the working fluid’s volumetric expansion ratio, mass flow rate, pressure ratio, density, temperature etc. A fluid with a higher volumetric expansion ratio would require a larger expander resulting in increased cost, weight and added packaging complexities [89].

Water is the working fluid traditionally used in power plant Rankine cycles. Water has advantages such as being environmentally friendly, non-toxic, cheap, good heat transfer characteristics and offering high cycle efficiency. However, issues such as condensation temperatures below ambient pressure bring in added complexity of improving fitting design to prevent air infiltration into the system. Water also requires
high pressure ratio’s that necessitate multi-stage expanders and better components overall to handle the pressure. There is also an issue of freezing when ambient temperature goes below 0°C. The added expense of using a water based WHR systems may not be cost effective to be implemented in automobiles in many cases mainly due to higher upfront investment costs.

ORC thermal efficiency is more sensitive to the condensing temperature than evaporating temperature due to the thermodynamic properties of most refrigerants [76]. Therefore, heat rejection at the condenser must be as efficient as practically possible to reduce the condenser pressure. [89] discusses practical considerations with regard to heat rejection at the condenser for commercial trucks as follows: Heat can be rejected at the condenser to outside air either through a direct air-working fluid (WF) heat exchanger or indirectly through another closed loop fluid. The direct air-WF method would be more efficient but requires more WF due to longer piping requirements. Another impediment to this configuration is to guarantee a steady stream of air but without hindering the aerodynamics of the vehicle and this may be achieved by placing the condenser somewhere on the vehicle not exposed to the stream and air can be forced through the condenser with a fan but that requires additional power consumption and also fan power consumption is the square of air flow rate.

The indirect method reduces the usage of additional WF but brings in an overall system efficiency penalty. The additional heat exchangers required would also add to the cost and weight of the system. [89] suggests placing the condenser in front of the radiator which reduces the radiator’s performance but since commercial trucks have an ample factor of safety in their engine cooling systems to handle worst case scenarios it may not cause engine cooling degradation, but it may not be the case for passenger vehicles.
The placement of the evaporator in the exhaust system causes backpressure, penalizing engine performance with increased pumping losses. A requirement for an evaporator is to be of high effectiveness and as compact as possible. Evaporators with a high effectiveness would require more passages which increases backpressure. Therefore a balance must be struck with these two requirements [90].

1.5.2 ORC Control

The highly transient nature of exhaust gases due to varied loads exerted on the engine coupled with the highly non-linear nature of an ORC poses a challenge in designing effective control strategies for ORC’s. Currently available literature heavily rely on PI control strategies which are not very suitable for such a highly non-linear system that is subjected to disturbances and prone to modeling errors [91]. The slow transient response of the system compared to faster transients of the waste heat source also impose additional difficulties. Nonetheless, PI controls have been used with reasonable performance in many cases, including practical implementations [81, 92, 92–94]. In most cases the overall control architecture consists of two independent PI controllers, one to regulating water flow rate which influences evaporator temperature and the other to regulate expander speed which influences evaporator pressure. [92] modeled a non-recuperative ORC using R-123 as the working fluid with a 11.6L heavy duty diesel truck engine serving as the source of exhaust. The control system implemented here uses the previously discussed PI control strategy along with a expander bypass controller that prevents wet fluid from entering the expander. The author reported 3.63% of the exhaust energy was recovered when tested with the regular driving pattern of a bus.

In [81], the aforementioned control scheme was successfully implemented for a
non-recuperative steam Rankine cycle on-board a Honda passenger vehicle as a test case.

[91] used a LQR controller over a linearized version of a Moving Boundary Model for the heat exchangers of a ORC, reporting satisfactory results.

Research in areas such as non-linear controls and control of recuperative ORC’s seems to be absent, thus highlighting the need for more investigation in the field of ORC controls.

1.6 Objectives of the Thesis

The focus of this thesis will be to model an ORC exhaust heat recovery system designed for a light duty vehicle using the commercial 1-D simulation software GT-POWER and to optimize the system to two parameters: Power output $P_{\text{net}}$ and Exergy Destruction $\dot{E}_d$ using a multi-objective implementation of the Particle Swarm Optimization algorithm.

Chapter 2 will detail the process of constructing the GT-POWER model. The following will be discussed in this chapter, a brief introduction into the capabilities of GT-POWER, layout of the system, data required to build the model, process of setting up and calibrating the heat exchangers, assembling the model and calibrating the inputs to sustain the design conditions followed by a verification of the model. The operation of the ORC at off-design conditions will be analyzed and finally feed-forward maps for the control inputs will be generated to maintain the design point for a range of exhaust conditions.

Chapter 3 will focus on providing an introduction to the Particle Swarm Optimization algorithm. An overview of optimization algorithms will be provided followed
by a detailed explanation of the PSO and its evolution over the years. The Multi-
objective implementation of the PSO will be discussed and the explanation of the
MOPSO algorithm will be provided with an example.

Chapter 4 will discuss optimization of the ORC. An exergy analysis of the ORC
will be performed. The implementation of the MOPSO for optimizing power output
and exergy destruction in the ORC will be discussed. An analysis on the behavior
of the system components along the Pareto front will be performed. An operating
point will be suggested and finally, the same process will be carried out with different
exhaust conditions and the trends will be observed.
CHAPTER 2
MODELING OF THE ORGANIC RANKINE CYCLE
SYSTEM

2.1 Layout of System

The layout of the ORC to be modeled in this thesis is based on the plant diagram shown in Figure 2.1

Figure 2.1: ORC plant diagram

The system is similar to the one described in Section 1.4.1 but with three additional elements: a receiver/dryer unit between the condenser and the pump, a bypass
valve across the exhaust gas side of the evaporator and a centrifugal coolant pump for the condenser. The coolant fluid for the condenser is water at 15°C. The exhaust is from a 3.6L V6 gasoline engine usually used in light duty pick-up trucks.

The control inputs to the system shown in Figure 2.1 are the feed pump speed, the expander speed and the bypass valve position on the exhaust gas side of the evaporator. The feed pump speed regulates the refrigerant mass flow rate and has a major effect on the evaporator outlet temperature. The expander speed controls the vapor volumetric flow rate and has an effect on the evaporator pressure. The exhaust bypass valve controls the heat input into the system and influences both the evaporator pressure and temperature.

A brief description of each component will be given below along with the data available to model them.

**Heat Exchangers**  All three heat exchangers in the current ORC system are plate heat exchangers (PHE) with the evaporator and condenser being two phase and the recuperator a single phase heat exchanger. A schematic of a simple configuration of a PHE is shown in Figure 2.2. A PHE consists of a series of flat plates pressed together and the fluids pass through alternate passages formed between the plates which act as the heat transfer surfaces. The fluids are delivered into and evacuated out of these passages by channels passing through the corners of the plates. To prevent inter-mixing of fluids when passing through the channels, insulating material is used around the channel openings in between alternating plates to prevent fluid from entering the wrong passage.

Plate heat exchangers have advantages such as compact size, low fouling tendency and superior heat transfer capabilities due to the flat shape of the heat transfer surface which is further aided by corrugations on the plates usually in a chevron pattern.
Plate heat exchangers of the gasketed variety also have the advantage of being convenient to disassemble and hence their use food processing industry where cleanliness is of utmost importance and the ease at which PHE’s can be disassembled is a huge advantage. Improved manufacturing technology and computational advances have also allowed better and more intricate corrugation designs in the plates improving PHE performance significantly. Brazed and welded type PHE’s are usually used for low heat load application such as the case of the ORC and the gasketed types are usually reserved for larger installations. Gasketed PHE’s are limited in their pressure and temperature handling capabilities.

To properly characterize the performance of the heat exchangers in GT-POWER three different set of data are required for the heat exchanger models, namely, geometric data, heat transfer data and pressure drop data.

Geometric data given by the suppliers includes the following parameters. The dry
mass of the heat exchanger influences transient behavior by its appearance in Equation 2.3.9 as $\rho V$ where it affects the rate of change of temperature. *Hydraulic diameter* also referred to as the characteristic length; and the *free flow area* are required to calculate Reynolds number that has an effect on the heat transfer characteristics (Equation 2.3.7). *Heat transfer area* is used in Equation 2.3.8 to calculate the heat transfer rate. *Number of fluid passages* is a multiplier on the hydraulic diameter, heat transfer and fluid flow area. *Fraction of volume in heat exchanger core* specifies the ratio of volume occupied by the tanks of the heat exchanger and appears in Equation 2.3.1. *Volume of fluid in the heat exchanger* influences the filling dynamics of the heat exchanger.

Heat transfer performance data is necessary for the calibration of heat transfer models. The performance of the heat exchanger will not be constant through all operating conditions, but rather will vary depending on the inlet temperature, pressure and mass flow rate. Ideally, performance data spread over the entire operating range of the heat exchanger will be required to model the heat transfer behavior and accurately predict the heat transfer and outlet temperatures.

Heat transfer in a heat exchanger can be calculated by any one of the following equations:

$$Q = \dot{m}_c \, c_{p,c} (T_{c,o} - T_{c,i}) \quad (2.1.1a)$$

$$Q = \dot{m}_h \, c_{p,h} (T_{h,i} - T_{h,o}) \quad (2.1.1b)$$

Excluding specific heat $c_p$, of all the other 7 variables in the two equations above atleast 6 must be known for different combinations of inlet fluid $\dot{m}$, $T$ and $p$ to construct a performance map that can model the heat exchanger. The performance of a heat exchanger can be denoted by its effectiveness which varies based on the inlet
conditions. It is defined as:

\[ \varepsilon = \frac{T_{h,i} - T_{h,o}}{T_{h,i} - T_{c,i}} \quad \text{(when } c_{p,h} < c_{p,c}) \] (2.1.2)

where \( 0 \leq \varepsilon \leq 1 \). Considering the definition of effectiveness, Equation 2.1.1 can also be expressed as:

\[ Q = \dot{m}_h c_{p,h} (T_{h,i} - T_{c,i}) \varepsilon \] (2.1.3)

When a heat exchanger is tested on a flow bench, pressures across all inlets and exits are also measured. The pressure is required to calculate \( c_p \) that is used in Equations 2.1.1 and also model the pressure drop across the heat exchanger. This can be used to predict losses attributed to fluid friction with the walls and expansion/contraction losses at the inlet and exit of the heat exchangers.

**Feed Pump** The feed pump chosen for the system is a diaphragm type positive displacement machine. The main advantages of positive displacement pumps is that there is almost no degradation in performance over a wide range of flow rates and also insensitivity to pressure differences. One drawback of the diaphragm pump is that they generate pulsations in the system which have to be neutralized with the use of a dampener.

**Expander** A positive displacement hermetically sealed scroll expander with a swept volume of 10cc is used in the ORC. Scroll expanders have advantages such a lesser moving parts due to the absence of valves and also a simple operating principle. They are also able to handle higher pressure ratio’s, lower operating speeds which makes it convenient in power conversion and component life; and higher resistance to two-phase flows. The isentropic efficiency of the expander is given by:

\[ \eta_{is} = \frac{h_4 - h_5}{h_4 - h_{5s}} \] (2.1.4)
where $h_{5s}$ is the ideal isentropic enthalpy at the expander exit. The isentropic efficiency is very dependent on the pressure ratio because effects of over-expansion and under-expansion degrade the isentropic efficiency significantly. Over-expansion is when the pressure inside the expander is lesser than the exit pressure and under-expansion is when it is higher. The power output of the expander can be expressed as:

$$P_e = \dot{m}_{ref} (h_4 - h_5) \tag{2.1.5}$$

GT-SUITE extends the data points from the lowest available pressure ratio to the pressure ratio of 1.

**Coolant pump** The coolant pump is a centrifugal pump. A constant pressure difference of 1 bar is assumed in the coolant circuit. The performance maps of the centrifugal pump are shown in Figure 2.3.

![Coolant pump performance maps](image)

(a) Speed map (rpm)  
(b) Efficiency map (%)

Figure 2.3: Coolant pump performance maps

**Receiver/Dryer** The receiver/dryer serves as a reservoir to store excess refrigerant in the system when not required by the process. It provides the system with a constant supply of refrigerant during transient operation of the ORC and also ensures
flow of sub-cooled fluid into the feed pump, thereby preventing cavitation. It also contains desiccant to remove any moisture that may have infiltrated the system. The receiver/dryer in the model is set to a volume of 3 liters.

### 2.2 Introduction to GT-SUITE

GT-SUITE [95] is a commercial 1-D fluid dynamics simulation software created by Gamma Technologies, Inc. with tools included for engine and vehicle simulations. In recent times, a package for vapor power and compression system simulation has been included, to allow for WHR system modeling.

In general, the software simulates the transient behavior of fluid systems by solving the conservation equations in a 1-D case:

**Continuity:** \( \frac{dm}{dt} = \sum_{\text{boundaries}} \dot{m} \quad (2.2.1) \)

**Energy (explicit solver):** \( \frac{d(me)}{dt} = -p \frac{dV}{dt} + \sum_{\text{boundaries}} (\dot{m}H) - hA_s(T_{\text{fluid}} - T_{\text{wall}}) \quad (2.2.2) \)

**Energy (implicit solver):** \( \frac{d(\rho HV)}{dt} = V \frac{dp}{dt} + \sum_{\text{boundaries}} (\dot{m}H) - hA_s(T_{\text{fluid}} - T_{\text{wall}}) \quad (2.2.3) \)

**Momentum:** \( \frac{d(\rho u)}{dt} = \left( \frac{dp}{dx} + \sum_{\text{boundaries}} (\dot{mu}) - 4C_f \frac{\rho u |u|}{2} \frac{dx}{D} - C_p \left( \frac{1}{2} \rho u |u| \right) A \right) \quad \frac{dx}{dt} \quad (2.2.4) \)
where

\[ \dot{m} \] boundary mass flux into volume
\[ m \] mass of fluid within the volume
\[ V \] volume
\[ p \] pressure
\[ \rho \] density
\[ A \] cross sectional flow area
\[ A_s \] heat transfer surface area
\[ e \] total internal energy per unit mass
\[ H \] Total enthalpy
\[ u \] velocity at the boundary
\[ C_f \] skin friction coefficient
\[ C_v \] pressure loss coefficient
\[ D \] equivalent diameter
\[ dx \] discretization length
\[ dp \] pressure difference across the discretization length

The above equations are generally applied to a staggered grid (Figure 2.4), and can be solved using an explicit or implicit solver. The staggered grid takes its name from the fact that the values of the scalar variables are taken as the average of a subvolume while the vector variables are calculated at the boundary of the subvolumes. The explicit solver (time marching) uses the conservation equations to calculate the derivatives of the properties for each subvolume at every time step. The derivatives are then used to determine the properties of the subvolume based on their values in the previous time step. The explicit solver is useful in simulating wave dynamics and fast transients encountered in engine simulations whereas the implicit solver
cannot do so. The explicit method’s maximum time step is limited by the Courant-Friedrichs-Lewy (CFL) condition which is a necessary condition for solver stability. The courant number (Equation 2.2.5) becomes the limiting factor on the timestep, based upon:

\[
\text{Courant number} = \frac{\Delta t}{\Delta x} (|u| + c) \leq 0.8 \quad (2.2.5)
\]

where

- \( \Delta t \) solver timestep
- \( \Delta x \) minimum discretization length
- \( u \) fluid velocity
- \( c \) speed of sound

The implicit solver is not as sensitive to the CFL condition and can have Courant numbers higher than 0.8. This allows the usage of timesteps significantly larger than the explicit method. Though the implicit solver requires multiple iterations to converge to a solution for every timestep, the performance gain in using a larger timestep far outweighs the penalty incurred due to the iterative process. Hence, this solver becomes useful to simulate the WHR system where slow transients dominate and wave dynamics are not important.

Figure 2.4: Staggered grid [10]
The implicit solver is used in simulating the ORC since the computation time is much faster and simulating wave dynamics is not important.

### 2.3 Heat Exchanger Models Setup and Calibration

The three heat exchanger models for the ORC system need to be individually calibrated before being integrated into the plant model. A standalone model for the evaporator, recuperator and condenser was created separately and calibrated to match the performance data provided by the supplier. GT-SUITE has calibration shell models requiring the user to only fill in the parameters. The calibration shells can be found in File→Open→Examples→Engine Waste Heat Recovery.

The standalone model is shown in Figure 2.5. The heat exchanger has two components, HxMaster and HxSlave parts connected to each other by a PointerConn part. They can be used interchangeably. HxMaster and HxSlave have to be connected to inlet and outlet pipes. The other end of the inlet pipes needs to be connected to an EndFlowInlet part or an EndFlowInletRefrig part to impose the fluid mass flow rate. The other ends of outlet pipes have to be connected to the EndEnvironment or EndEnvironmentRefrig part to impose the pressure in the circuit.

A RefrigCircInit part must be used for each refrigerant circuit in the model. The RefrigCircuitInit part initializes the fluid state for that circuit. The Part Name Object Identifying Refrigerant Circuit to be Initialized attribute of the RefrigCircuit part allows the user to link a refrigerant circuit to it. The User Specified Average System Density was set to 300 kg/m$^2$ as recommended by GT-SUITE. It must also be noted that the fluid, heat exchanger and pipe wall temperatures must be initialized with a common temperature to avoid solver convergence errors at the beginning of the simulation. 300 K was chosen to be the initial temperature.
All the properties of the heat exchanger need to be entered in HxMaster. The options available within HxMaster are in Figure 2.6. A HeatExchangerSpecs template object must be created to be linked in the As Tested Heat Exchanger Specifications Object attribute. HeatExchangerSpecs template allows entry of geometric, heat transfer and pressure drop data of the heat exchanger. The Discretization length attribute is used to specify the length of each sub-volume in the heat exchanger (See 2.3.1).

![Figure 2.5: Evaporator standalone model](image)

![Figure 2.6: Evaporator HxMaster dialog box](image)

The HeatExchangerSpecs template dialog box appears as in Figure 2.7. Within this template four separate objects for geometric, heat transfer and Master and Slave pressure drop have to be created. The tree in Figure 2.8 shows all the templates that have been used by one of the three heat exchangers.
2.3.1 Geometric Data

The geometric data object to be used will depend on the amount of data available and the type of heat exchanger. GT-SUITE has templates specific to Tube fin, Plate and Shell Tube type heat exchangers where attributes specific to each of these heat exchangers can be entered. If detailed geometric data is not available or if the heat exchanger is not one of these types then a general template object called HxGeomGeneral is available where just basic data can be entered. HxGeomGeneral template was used for all heat exchangers in the WHR system. The geometric properties of the heat exchanger have to be entered in this template.

GT-SUITE divides the heat exchanger into multiple sub-volumes internally as shown in 2.9. The equations for the heat transfer, fluid state and pressure drop are
all calculated for each sub-volume. It can be noted from Figure 4 that the masses are not connected to each other directly, hence there is no axial heat transfer between the walls of each sub-volume. The following equations are used for discretization:

\[
\text{No. of subvolumes} = \frac{\text{HX Length}}{dx} = \frac{\text{Master HX Vol.} \times \text{Frac. of Vol. in HX core}}{\text{Master Flow Area}} \times \frac{1}{dx} \quad (2.3.1a)
\]

\[
\text{HT area per subvolume} = \frac{\text{HT area} \times \text{No. of identical tubes}}{\text{No. of Subvolumes}} \quad (2.3.1b)
\]

\[
\text{Flow area per subvolume} = \text{Flow area} \times \text{No. of identical tubes} \quad (2.3.1c)
\]

The discretization length was chosen to be 35mm for all heat exchangers as a compromise between modeling accuracy and computation time.

![Heat exchanger internal structure](image)

**Figure 2.9: Heat exchanger internal structure**

### 2.3.2 Heat Transfer Models

The heat transfer for two-phase fluids can be modeled in GT-SUITE by either predictive correlation models or Nusselt-Reynolds generic correlation. The former allows the selection of a separate correlation model (see Figure 2.10) for each phase of the refrigerant. The Nusselt-Reynolds correlation method (See page 40) calculates the
heat transfer coefficient using a general empirical equation and calibrating tuning parameters to match the heat exchanger performance data provided by the supplier. The heat transfer for single phase fluids, such as the exhaust and coolant must be modeled only using the Nusselt-Reynolds correlation method in GT-SUITE.

GT-SUITE has different options towards selecting a suitable heat transfer correlation for each phase in the case of two-phase fluids.

For liquid and vapor phases, the Dittus-Boelter correlation (Equations 2.3.2a and 2.3.2b) is used. It is a relatively simple and robust equation to estimate the heat transfer coefficient:

\[
h = 0.023 \frac{Re^{0.8} Pr^{0.4} k}{D} \quad \text{(Heating)} \quad (2.3.2a)
\]

\[
h = 0.023 \frac{Re^{0.8} Pr^{0.3} k}{D} \quad \text{(Cooling)} \quad (2.3.2b)
\]

According to [96] the Dittus-Boelter correlation is valid for fully developed flow \((L/D \geq 10)\), \(Re \geq 10,000\) and \(0.7 \leq Pr \leq 160\).

Plate Heat Exchangers (PHE) have different heat transfer characteristics when compared to heat transfer in pipe flow [97, 98] as in shell and tube heat exchangers; and have only become popular very recently. Hence, availability of literature on PHE heat transfer characteristics is very limited, especially for two-phase flow. For this reason, GT-SUITE has just one recommended heat transfer correlation for boiling and condensation in PHE’s.

For refrigerant boiling, the Yan-Lin correlation [97] is used. This correlation was formulated from experimental results obtained using R-134a as the refrigerant in a PHE.

\[
h = 1.926 Re_{eq}^{1/3} Pr^{-0.5} B_{eq} k_{l} \frac{k}{D_h} \quad (2.3.3)
\]
where

\[ R_{eq} = \frac{G_{eq} D_h}{\mu_l} \quad \text{Bo}_{eq} = \frac{q''_w}{G_{eq} i_{fg}} \] (2.3.4)

\[ G_{eq} = G \left[ (1 - X_m) + X_m \left( \frac{\rho_l}{\rho_g} \right)^{0.5} \right] \] (2.3.5)

- \( k_l \) conductivity of liquid phase
- \( D_h \) hydraulic Diameter
- \( X_m \) fluid quality
- \( i_{fg} \) heat of vaporization
- \( q''_w \) wall heat flux

and \( R_{eq} \) and \( \text{Bo}_{eq} \) are equivalent Reynolds and Boiling numbers respectively as in [99]. The authors reported the correlation fit the experimental results with an average deviation of 8.3%.

For condensation the Yan-Lio-Lin correlation [100] is used. This correlation was also formulated based upon experimental results obtained using R-134a, leading to a reported correlation accuracy within 15%:

\[ h = 4.118 \, R_{eq}^{0.4} \, Pr^{1/3} \, \frac{k_l}{D_h} \] (2.3.6)

where

\[ \text{Bo} = \frac{q''_w}{G i_{fg}} \]

Unlike the predictive model, which requires specific information on the fluid, GT-
SUITE allows one to approximate the heat transfer coefficient using the generic Nusselt-Reynolds correlation:

$$Nu = C \, Re^m \, Pr^{1/3}$$

(2.3.7)

where

$$Nu = \frac{hL}{k}$$

$$Re = \frac{\rho U L}{\mu}$$

$$Pr = \frac{\mu c_p}{k}$$

$$U = \frac{\dot{m}}{\rho A_f}$$

$h$ heat transfer coefficient
$L$ characteristic length
$k$ thermal conductivity of the fluid
$\rho$ density of the fluid
$U$ fluid mean velocity
$\mu$ dynamic viscosity
$c_p$ specific heat of the fluid
$\dot{m}$ mass flow rate of the fluid
$A_f$ flow cross sectional area

$C$ and $m$ parameters for each flow regime must be calibrated to fit the experimental data available. These parameters, along with the transition points between the flow regimes, will be automatically tuned by GT-SUITE using a non-linear regression fit. The heat transfer coefficient is then used in Equation 2.3.8 to determine the heat transfer between the wall and the fluid.

There are three templates available for entering heat transfer data as enumerated
below. These templates must be linked to the *Heat Transfer Data Object* attribute in the **HeatExchangerSpecs** template.

1. **HxNuMap**
2. **HxNuMapRefrig**
3. **HxHeatTransferCorr**

**HxNuMap** is used to enter performance data obtained from experiments for heat exchangers that use single phase fluids only and will not be used in this model since the refrigerant (a two-phase fluid) flows through all the heat exchangers used in this model.

**HxHeatTransferCorr** has two attributes *Internal (Master) Heat Transfer Correlation* and *External (Master) Heat Transfer Correlation*. For the calculation of heat transfer coefficient, these attributes allow selection of either the predictive correlation method or Nu-Re method for both the Master and Slave sides of the heat exchanger by selecting the appropriate template. **HxHTCorrRefrig** template object is used to specify the predictive correlation equations and also allows the use of multipliers on the heat transfer coefficients to adjust wall temperature and heat transfer rate. In the **HxNusseltCorr** template The Nu-Re correlation parameters, can either be manually entered (if already available) or entered using values output from the regression analysis performed when using **HxNuMapRefrig** (see below). This manual entry of Nu-Re paramters allows the user to tweak the parameters output from the regression done by the **HxNuMapRefrig** template object, if the results do not match data satisfactorily.

**HxNuMapRefrig** must be used for two phase fluids. In this template, the user can select to use the Nusselt-Reynolds correlation on both the Master and Slave
side or just on one side. GT-SUITE suggests using a two-sided Nu-Re correlation for
single phase heat exchangers such as recuperators that have a two phase fluid
flowing through them. For two phase heat exchangers, a predictive correlation for
the refrigerant side and the Nu-Re correlation for the other side. If using a one-sided
Nu-Re correlation then the template requires an object of the HxHTCorrRefrig
template to specify the correlation scheme to be followed in the other side of the heat
exchanger.

The heat transfer coefficient on both sides of the heat exchanger is first calculated
using the methods outlined before, then the total heat transfer between the fluid and
the wall is calculated by:

\[ q = hA\Delta T \]  \hspace{1cm} (2.3.8)

where

- \( q \) heat transfer through the wall
- \( h \) heat transfer coefficient
- \( A \) heat transfer area
- \( \Delta T \) temperature difference between the wall and the fluid

If there is an imbalance in the heat transfer between both sides of the heat ex-
changers, the difference goes into heating or cooling the heat exchanger walls:

\[ \frac{dT_{\text{wall}}}{dt} = \frac{Q_M + Q_S}{\rho VC_p} = \frac{(hA\Delta T)_M + (hA\Delta T)_S}{\rho VC_p} \]  \hspace{1cm} (2.3.9)
where:

- $Q_M$: wall-fluid heat transfer on the master side
- $Q_S$: wall-fluid heat transfer on the slave side
- $\rho$: density of the wall material
- $V$: volume of the wall material
- $C_p$: specific heat capacity of the wall material
- $h$: heat transfer coefficient
- $A$: heat transfer area

### 2.3.3 Pressure Drop Data

Pressure losses in a heat exchanger can be attributed to contraction/expansion of the fluid entering and exiting the heat exchanger manifolds and friction in the micro channels.

Friction is accounted for in the momentum equation using the Darcy-Weisbach equation [101]:

$$h_f = 4C_f \frac{L v^2}{D 2g} \quad (2.3.10)$$

where

- $h_f$: head loss $= \frac{p}{\rho g}$
- $C_f$: Fanning friction factor
- $L$: length of tube
- $D$: Diameter of tube
- $v$: fluid velocity
- $g$: acceleration due to gravity

For laminar flow, the fanning friction factor ($C_f$) is determined by equating the
Hagen-Poiseulle equation with the Darcy-Weisbach equation, which for a smooth pipe
(roughness = 0), results into:

\[ 4C_f \frac{L v^2}{D 2g} = \frac{32 \mu L v}{\rho g D^2} \]

\[ C_f = \frac{16 \mu}{\rho v D} \]

\[ C_f = \frac{16}{Re} \quad (Re < 2000) \quad (2.3.11) \]

For turbulent flow, two methods exist in GT-SUITE. The simpler method uses the
Blausius correlation (Equation 2.3.12).

\[ C_f = \frac{0.08}{Re^{0.25}} \quad (Re > 4000) \quad (2.3.12) \]

A more detailed method uses Serghides explicit approximation [102] of the Colebrook-
White equation [103] which models the Moody diagram:

\[ C_f = \frac{1}{4} \left(4.781 - \frac{A - 4.781}{B - 2A + 4.781}\right)^{-2} \quad (Re > 4000) \quad (2.3.13) \]

where

\[ A = -2 \log_{10} \left(\frac{\varepsilon}{3.7D} + \frac{12}{Re}\right) \]

\[ B = -2 \log_{10} \left(\frac{\varepsilon}{3.7D} + \frac{2.51A}{Re}\right) \]

\[ \varepsilon: \text{ pipe wall sand roughness height} \]

The simple or improved method can be chosen by going to the Run Setup dia-
log box (Run→Run Setup), selecting the FlowControl tab and in the Flow Settings
Object for Real Gas, Friction, Heat Transfer attribute. A FlowControlSettings template object will be present in this attribute. The Flow Loss Model attribute within the FlowControlSettings object allows toggling between the simple and improved method. Using the improved method increases computation time but provides higher accuracy.

The coefficient of discharge \((C_p)\) is defined as:

\[
C_p = \frac{p_1 - p_2}{\frac{1}{2} \rho_1 v_1^2}
\]  

(2.3.14)

where

\(p_1\)  inlet pressure
\(p_2\)  exit pressure
\(\rho_1\)  inlet density
\(v_1\)  inlet velocity

Three templates have been used in calibrating the pressure drop in the heat exchangers as mentioned below. These templates must be linked to the Internal (Master) Pressure Drop Data Object and External (Slave) Pressure Drop Data Object attributes in the HeatExchangerSpecs template:

1. FlowPDropHxMapRefr
2. FricCorrRefrig
3. FlowPdropUser
**FlowPDropHxMapRefr**  This template can be used to model the pressure drop for two-phase fluids, taking the pressure data from the **HxNuMapRefrig** template object that was used to specify heat transfer characteristics and automatically calibrates the friction multipliers and orifice $C_d$ to match data. Within this template, there is a *Pressure Drop Data Object* attribute that requires entering a **FricCorrRefrig** template object that has options to choose the predictive friction correlation methods to use. **FricCorrRefrig** template also allows for entering multipliers for the friction coefficient for each fluid phase but since GT-SUITE automatically calibrates the friction multipliers, manually entered multipliers in this template will be ignored.

This is the easiest way to model pressure drop since it requires no calibration on the user’s side but in many cases the calibration is not satisfactory and manual adjustment using the following two templates has to be done.

**FricCorrRefrig**  If the previous template does not accurately model the pressure drop, then this template can be tried next. This template is used to only model pipe friction ignoring the discharge coefficient. The friction model can be chosen and friction multiplier can be manually tuned to match the data.

**FlowPdropUser**  This template can be used if the above two templates fail to provide a reasonable fit to the pressure drop data. This template allows manually setting the discharge coefficient and the friction multiplier. Both these values can be tuned either manually or through an external optimization routine to match pressure drop data.
2.3.4 Calibration of the Evaporator Model

The evaporator is a two-phase heat exchanger using exhaust gas as the fluid in the external circuit. **HxMaster** was used for the refrigerant side. An object for the evaporator using the **HeatExchangerSpecs** template was created. Geometric data was entered using the **HxGeomGeneral** template.

Evaporator performance data was entered into an object created from **HxNuMapRefrig** template. The performance data was split into two sets in a 60/40 split with the former being used to calibrate the model and the latter set used for validating the performance of the calibrated heat exchanger.

Predictive correlations were used on the refrigerant side and Re-Nu correlation was used for the exhaust gas side. A **HxHTCorrRefrig** template object has to be created in the **Master (Internal) Refrigerant Heat Transfer Correlation** attribute to specify the type of predictive correlations to use on the refrigerant side. The heat transfer correlations chosen for each phase in the HxHTCorrRefrig object are shown in Figure 2.10.

![Figure 2.10: Predictive correlations for each phase of the refrigerant](image)

For the pressure drops, the **FlowPDropHxMapRefr** and **FlowPdropUser** templates were used for the refrigerant and exhaust gas sides respectively. The friction multiplier and discharge coefficient on the exhaust gas side were tuned manually until a reasonable fit was obtained.

The calibration result plots are shown in Figure 2.11 and the error for each plot

48
is in Table 2.1. The exhaust gas exit temperature had an offset with the data that can be attributed to the different specific heat \( c_p \) of the exhaust gas between the data and the GT-SUITE model. A correction factor was applied by taking the ratio of specific heats from the two data sets.

![Evaporator calibration correlation plots](image)

(a) Heat rate  
(b) Refrigerant outlet temperature  
(c) Exhaust gas outlet temperature  
(d) Refrigerant pressure drop

Figure 2.11: Evaporator calibration correlation plots

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Mean Error</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat rate</td>
<td>0.83</td>
<td>%</td>
</tr>
<tr>
<td>Refrigerant Exit Temperature</td>
<td>2.72</td>
<td>K</td>
</tr>
<tr>
<td>Exhaust gas exit Temperature</td>
<td>7.01</td>
<td>K</td>
</tr>
<tr>
<td>Pressure drop across refrigerant side</td>
<td>0.1</td>
<td>bar</td>
</tr>
</tbody>
</table>

Table 2.1: Evaporator calibration error summary
2.3.5 Calibration of the Recuperator Model

The recuperator is a single phase heat exchanger with liquid refrigerant on the cold side and vapor on the hot side. The cold side uses the HxMaster part and the hot side the HxSlave part. Entering geometric data is similar to the evaporator model. The pressure drops were modeled for both sides using the FricCorrRefrig template, where the friction multipliers were manually calibrated.

The heat transfer was modeled using the Nusselts-Reynolds correlation on the gas side and the predictive correlation on the liquid side.

The results of the calibration are summarized in Figure 2.12 and the error is tabulated in Table 2.2.

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Mean Error</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat rate</td>
<td>4.4</td>
<td>%</td>
</tr>
<tr>
<td>Cold side exit temperature</td>
<td>1.81</td>
<td>K</td>
</tr>
<tr>
<td>Hot side exit temperature</td>
<td>4.5</td>
<td>K</td>
</tr>
<tr>
<td>Pressure drop across Cold side</td>
<td>0.02</td>
<td>bar</td>
</tr>
<tr>
<td>Pressure drop across Hot side</td>
<td>8.6e-4</td>
<td>bar</td>
</tr>
</tbody>
</table>
(a) Heat rate

(b) Cold side outlet temperature

(c) Hot side outlet temperature

(d) Cold side pressure drops

(e) Hot side pressure drops

Figure 2.12: Recuperator calibration correlation plots
2.3.6 Calibration of the Condenser Model

The condenser is a two-phase heat exchanger with water being the fluid in the external circuit. The refrigerant is modeled using the HxMaster part and the water uses a HxSlave part. The Condenser performance modeling is also identical to the evaporator with the refrigerant side using the predictive correlations and the Nusselt-Reynolds correlation being used for the coolant side.

The pressure drops were modeled using the FlowPdropUser template since the other methods were not adequate in this specific case.

The calibration results are shown in Figure 2.13 and the error for each plot is given in Table 2.3.

![Figure 2.13: Condenser calibration correlation plots](image-url)

(a) Heat rate  
(b) Refrigerant outlet temperature  
(c) Refrigerant pressure drops  
(d) Coolant pressure drops

Figure 2.13: Condenser calibration correlation plots
Table 2.3: Condenser calibration error summary

<table>
<thead>
<tr>
<th>Quantity</th>
<th>Mean Error</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Heat rate</td>
<td>1.38</td>
<td>%</td>
</tr>
<tr>
<td>Refrigerant exit temperature</td>
<td>1.68</td>
<td>K</td>
</tr>
<tr>
<td>Pressure drop across refrigerant</td>
<td>6e-3</td>
<td>bar</td>
</tr>
<tr>
<td>Pressure drop across coolant</td>
<td>0.028</td>
<td>bar</td>
</tr>
</tbody>
</table>

2.4 Pump Model Setup

The PumpRefrig template part is required for the feed pump. It is a generic template that only requires the user to enter the maps. The data points for the feed pump maps provided by the supplier need to be entered in the PumpMap template object and this object is to be linked to the Reference Object or File attribute in the main tab of the PumpRefrig part. The pump speed is imposed externally with the SpeedBoundaryRot template part and linked to the PumpRefrig part. It has the Imposed Speed attribute in which the desired pump speed is entered.

The state of the fluid exiting the pump can be determined with the isentropic efficiency equation below:

$$\eta_{is} = \frac{h_2 - h_1}{h_2 - h_1}$$  \hspace{1cm} (2.4.1)

The coolant pump uses a Pump template object. This part differs from the PumpRefrig part in that it uses ideal gas laws to calculate the isentropic enthalpy rather than the fluid property tables which reduces computational time. The setup process is similar to that of the feed pump in all other respects.
2.5 Expander Model Setup

The expander model uses the TurbineRefrig template object. The data points for the expander maps have to be entered in the TurbineMapGridRaw template object. The refrigerant type has to be selected in the main tab of this object and the datapoints entered in the data tab. Once this is done, the TurbineMapGridRaw object has to be linked to the Map Object or File attribute in the maps tab of the TurbineRefrig part.

From the maps, the flow rate and isentropic efficiency are determined from the pressure ratio and speed and output state of the map is calculated based on this.

2.6 Receiver Model Setup

The Receiver/Dryer model’s purpose is to send out liquid refrigerant even when it’s inflow is gaseous. It does so by storing a buffer of liquid fluid but the receiver can only supply as much liquid as it has stored. The Receiver model is setup with the ReceiverDryerRefrig template part. The volume of the Receiver can be set using Volume attribute in the main tab. In the options tab, the user is able to set the Liquid Volume Fraction (LVF) threshold below which the receiver starts sending out gaseous refrigerant.

\[
LVF = \frac{V_{\text{liquid}}}{V_{\text{liquid}} + V_{\text{vapor}}}
\]

2.7 Assembly & Verification

After calibrating the heat exchangers, the model is assembled by connecting different components together using pipe elements. The PumpRefrig and TurbineRefrig template parts are used for the feed pump and expander respectively. The Pump
template object is used for the coolant pump. Heat transfer and pipe friction was ignored in all pipes by setting the respective attributes in every pipe and connector to zero. The receiver capacity was set to 3 liters. The EndFlowInlet part was used as the source of exhaust gas. Rather than using a valve for the exhaust gas bypass, it was simulated by using a multiplier on the total exhaust gas mass flow rate exiting the exhaust.

![ORC GT-SUITE model](image)

Figure 2.14: ORC GT-SUITE model

### 2.7.1 Initialization Interface and Parameters

GT-SUITE allows the user to enter variables in fields of parts in the model and control the values of the variables in one convenient single window called the Case Setup
dialog box which can be accessed through Run→Case Setup. Parameters that are regularly modified such as exhaust temperature and mass flow rate; control inputs, etc. can be setup as variables in their respective parts by using the convention `variable_name` and accessed in the case setup. The user is able to run a parameter sweep of multiple variables through this interface. Other features present in the case setup are automating the entry of parameter values for multiple cases by setting simple equations that define the spacing intervals between each case and also a variable can be made dependent on another variable by referring to them in the equations.

Figure 2.15: GT-SUITE case setup dialog box

2.7.2 Solver Settings

Solver settings can be accessed in the Run Setup dialog box (Run→Run Setup). The relevant settings pertaining to the WHR will be described below.
TimeControl

The *Time Control Flag* attribute is set to **continuous**, to track all variables with respect to time as opposed to crank angle which occurs when the time control is set to **periodic** for engine simulations. The maximum simulation duration time can also be set in this tab. The duration set must long enough to ensure steady state convergence of the simulation. The *Improved Solution Sequence for Multi-Circuit Models* attribute is **unchecked**.

Initialization

This tab allows the user to set the initial conditions of the model when starting a simulation. Setting initial conditions sufficiently close to the final steady state values will reduce simulation time. If the *Initialization State* attribute in this tab is set to **previous case** the solver will use the final state of the previous case to initialize the next case. Another option is to import the initial state from the results of another simulation but this option is not reliable enough to be used in the current version of GT-SUITE.

FlowControl

*Part Name List Object Identifying Circuit Belonging to Column* attribute is used to label all the flow circuits in the model for easy identification in post processing. In the case of the ORC model, there are three separate circuits, namely exhaust gas, refrigerant and coolant. A **FlowCircPartSelector** template object has to be created and names for each circuit have to be assigned in this object. The *Solution method* attribute must be set to **implicit** as discussed previously. The *Time Step and Solution Control Object* attribute can be used to set the time step length, maximum iterations and other related settings. When this object is set to **def**, GT-SUITE will assumes default settings specific to WHR systems. The defaults can be changed using
a FlowControlImplicit template object. In this case the Number of Iterations and Maximum Number of Times to Reduce Time Steps attributes were set to 200 and 2 respectively to reduce the computation time.

<table>
<thead>
<tr>
<th>Setting</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of iterations</td>
<td>250</td>
</tr>
<tr>
<td>Maximum Number of Times to Reduce Time Step</td>
<td>3</td>
</tr>
<tr>
<td>Species Damping Factor</td>
<td>0.75</td>
</tr>
<tr>
<td>Energy Damping Factor</td>
<td>0.75</td>
</tr>
<tr>
<td>Pressure Relaxation Factor</td>
<td>0.75</td>
</tr>
<tr>
<td>Temperature Relaxation Factor</td>
<td>0.75</td>
</tr>
<tr>
<td>Density Relaxation Factor</td>
<td>0.75</td>
</tr>
</tbody>
</table>

**Thermal Control**

The Wall Temperature Solver attribute has to be set to transient for the solver to consider transients in the wall masses of pipes and heat exchangers.

**2.7.3 Model Tuning**

After the model was assembled, what remains is to decide the amount of refrigerant to add to the circuit, tune the control inputs and coolant flow rate to match the “design point” of ORC (see Table 2.7). More on the design point will be discussed in section 2.9.
First, the feed pump speed is determined. The approximate pump speed can be determined using Equation 2.7.1, assuming an ideal volumetric efficiency:

\[ \dot{m} = \rho V_d N \]  

(2.7.1)

where

- \( \rho \)  
  refrigerant density

- \( V_d \)  
  pump displacement

- \( N \)  
  pump speed

The heat transfer rates at the evaporator and condenser determine the evaporator and condenser pressure respectively. Hence by adjusting the exhaust bypass and coolant flow rate, the respective pressures can be adjusted to their desired values.

Attaching **SignalActiveDial** parts on each component makes it possible to adjust the pump speed, exhaust bypass, expander speed and coolant flow rates while the simulation is running to match the design point. This helps reduce the tuning process.

It is important to note that numerical simulations in two-phase flows are more susceptible to mass balance errors when compared to single phase systems. This is because of the vast difference in density in two-phase fluids during evaporation and condensation process. A non-convergent time step may leave the system in a state where the conservation equations have not been satisfied, causing a huge discrepancy in density at one or more sub-volumes resulting in the overall mass between timesteps diverging significantly. This happens during fast transients where the densities fluctuate rapidly. Gradually ramping the boundary conditions from initialization to the intended values might mitigate the mass balance error in most cases. These ramps can be set substituting the quantity with **ProfileTransient** template which defines the ramp based on time. Mass balance error of up to 1% was deemed acceptable
for the WHR system. See Figure 2.16 for the ramp profiles used for the boundary conditions.

Figure 2.16: Example of boundary condition initialization ramps to improve convergence

The amount of refrigerant is first roughly calculated using the volume and assumed densities of the gas side and liquid side of the circuit. The simulation is run with the calculated amount of mass. If the mass is underestimated, the model will output saturated vapor at the condenser outlet. Numerical oscillations at the outlet were also observed at the condenser outlet when there was insufficient mass. The sub-cooling at the condenser outlet will increase with increase in refrigerant mass. The amount of refrigerant was adjusted until the sub-cooling was found slightly more than 5 K (see Figure 2.17).
Figure 2.17: Effect of total refrigerant mass in the system on subcooling at the condenser outlet
2.8 System Inputs/Outputs and Constraints

The variables that affect the performance of the ORC are exhaust and coolant conditions and control inputs. The set of variables under each of these three groups is summarized in Table 2.5. The exhaust temperature and mass flow rate are primarily dependent on the engine speed and load. The source of water i.e. coolant for the condenser is assumed to be an infinitely large reservoir maintained at a constant temperature of 15°C and the flow rate through the condenser is regulated by a PI controller to maintain the target condenser pressure. The condenser pressure is strongly dependent on the heat rejection. Exhaust conditions and coolant temperature can be considered as disturbances but will remain constant for most of the analyses that follow. Control inputs to the system are the feed pump speed $N_{fp}$, expander speed $N_e$ and exhaust bypassed $x_{BP}$.

Since the model does not take into account real world effects and considerations such as pump cavitation and fluid degradation at high temperatures, a few boundaries have to be set on the operation of the control inputs and the states at various locations with the ORC plant model to prevent situations that might potentially damage the system. Table 2.6 lists the constraints that were imposed on the system states and control inputs.

<table>
<thead>
<tr>
<th>Exhaust</th>
<th>Control Inputs</th>
<th>Coolant</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exhaust Temperature</td>
<td>Feed Pump Speed</td>
<td>Coolant Temperature</td>
</tr>
<tr>
<td>Exhaust Mass Flow Rate</td>
<td>Expander Speed</td>
<td>Coolant Pump Speed</td>
</tr>
<tr>
<td>Bypass valve position</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2.5: Boundary variables for the ORC
Table 2.6: ORC physical constraints

<table>
<thead>
<tr>
<th>State</th>
<th>Limits</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>System:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refrigerant Mass Flow Rate</td>
<td>$0.02 &lt; \dot{m}_{ref} &lt; 0.07$</td>
<td>kg/s</td>
</tr>
<tr>
<td>Evaporator Pressure</td>
<td>$15 &lt; p_{evap} &lt; 25$</td>
<td>bar</td>
</tr>
<tr>
<td>Evaporator Outlet Temperature</td>
<td>$T_{evapout,sh} &lt; 235$</td>
<td>°C</td>
</tr>
<tr>
<td>Evaporator Inlet Subcool</td>
<td>$\Delta T_{evapin,sc} &lt; -5$</td>
<td>°C</td>
</tr>
<tr>
<td>Evaporator Outlet Superheat</td>
<td>$\Delta T_{evapout,sh} &gt; 5$</td>
<td>°C</td>
</tr>
<tr>
<td>Condenser Outlet Subcool</td>
<td>$\Delta T_{condout,sc} &lt; -5$</td>
<td>°C</td>
</tr>
<tr>
<td><strong>Control Input:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feed pump Speed</td>
<td>$175 \leq N_{fp} \leq 675$</td>
<td>rpm</td>
</tr>
<tr>
<td>Expander Speed</td>
<td>$1400 \leq N_{e} \leq 4400$</td>
<td>rpm</td>
</tr>
<tr>
<td>Exhaust Bypass Position</td>
<td>$0 \leq x_{BP} \leq 1$</td>
<td>fraction</td>
</tr>
</tbody>
</table>

Refrigerant mass flow rate $\dot{m}_{ref}$ range is limited by flow rate capacity of the expander as per specifications provided by the manufacturer. Feed pump speed $N_{fp}$ has a linear relationship with $\dot{m}_{ref}$ due to it being a positive displacement machine and this property can be exploited to set the range of operation for $N_{fp}$ to correspond to the flow capacity of the expander. The range of the expander speed $N_{e}$ is set as per manufacturer specifications. Refrigerant flow at the evaporator outlet has to be in vapor phase to prevent damage to the expander since the presence of fluid greatly accelerates degradation of the expander seals. No phase change should be allowed to happen in the recuperator outlets to prevent overheating of all heat exchangers because premature phase change in the recuperator will result in higher wall temperatures due to changes in heat transfer coefficients causing undue stress.
on the heat exchangers reducing their operating life. Maximum pressure is limited by the capacity of the heat exchangers. The refrigerant undergoes decomposition at temperatures over 250°C [104] and naturally it should not be allowed to go over this limit at any cost. The maximum temperature at the evaporator outlet has been restricted to 235°C for this reason with a 15°C safety buffer provided. Condenser outlet has to be subcooled to prevent cavitation at the pump.

2.9 The Design Point

The “Design point” is an operating condition for the ORC designed to provide reasonable performance and that can also be sustained for a fairly wide range of exhaust gas conditions around the typical highway cruising speed of 65 mph where the exhaust flow is relatively constant to allow steady operation of the system. The design point will be used as the reference condition when optimizing the ORC. The exhaust temperature and mass flow rate corresponding to 65 mph are 651°C and 32.5 g/s respectively. With this energy source in mind, target values for four states of the ORC, evaporator outlet temperature ($T_3 = 210°C$), refrigerant mass flow rate ($\dot{m}_{ref} = 0.04 \text{ kg/s}$), evaporator and condenser pressure ($p_H = 21 \text{ bar}$ and $p_L = 2.5 \text{ bar}$ respectively) were chosen. Consideration was also given to heat exchanger performance, material and refrigerant physical limits; and cost impact on components while deciding the design point. The pressures $p_H$ and $p_L$ were chosen to coincide with the manufacturer recommended pressure for the expander where the isentropic efficiency is at its maximum. Another major factor in the selection of the design point was its ability to be sustained over a wide range of exhaust conditions, since they will not be constant during vehicle operation.

The combination of control inputs obtained that correspond to these target design conditions are in Table 2.7 along with all other the relevant values of the ORC at
design point. Figure 2.19 is a reproduction of Figure 1.12 that shows the T-s and p-H thermodynamic diagrams for the design point with thermodynamic information also provided on the states at each point. Figure 2.1 is reproduced as Figure 2.18 with labels added for the states of various points in the system.

Figure 2.18: ORC plant diagram
Table 2.7: Characterization of Design Point

**Vehicle States:**
- Vehicle Speed: 65 mph
- Exhaust Temperature: 651°C
- Exhaust Mass Flow Rate: 0.0325 kg/s

**Exhaust gas bypassed**: 46%

**ORC Target States:**
- Evaporator Outlet Temp.: 483°C
- Refrigerant Mass Flow Rate: 0.04 kg/s
- Evaporator Pressure: 21 bar
- Condenser Pressure: 2.5 bar

**Control Input Positions:**
- Pump Speed: 365 RPM
- Expander Speed: 2933 RPM
- Exhaust gas bypassed: 46%

**Coolant Pump:**
- Coolant Pump Speed: 4060 RPM
- Coolant Flow Rate: 104.2 g/s

---

<table>
<thead>
<tr>
<th>State Point</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pressure</td>
<td>2.5</td>
<td>21</td>
<td>21</td>
<td>21</td>
<td>2.5</td>
<td>2.5</td>
<td>bar</td>
</tr>
<tr>
<td>Superheat/Subcool</td>
<td>-9</td>
<td>-93</td>
<td>-18</td>
<td>85</td>
<td>122</td>
<td>17</td>
<td>K</td>
</tr>
<tr>
<td>Temperature</td>
<td>304</td>
<td>305</td>
<td>379</td>
<td>483</td>
<td>435</td>
<td>330</td>
<td>K</td>
</tr>
</tbody>
</table>

Figure 2.19: T-s and p-H diagram for an ORC Cycle
A first law analysis is done on the design point to understand energy flow in the system. This also allows us to verify proper working of the model. The energy balance equation is as follows:

\[
\dot{Q}_{E1} + \dot{Q}_{C1} + P_{fp} + P_{cp} = \dot{Q}_{Eb} + \dot{Q}_{E3} + \dot{Q}_{C3} + P_e
\]  

(2.9.1)

<table>
<thead>
<tr>
<th>Inputs</th>
<th>Component %</th>
<th>Outputs</th>
<th>Component %</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exhaust</td>
<td>76.8000</td>
<td>Exh. bypassed</td>
<td>35.320</td>
</tr>
<tr>
<td>Coolant in</td>
<td>22.7000</td>
<td>Evap. exh. out</td>
<td>5.828</td>
</tr>
<tr>
<td>Feed pump</td>
<td>0.2681</td>
<td>Expander</td>
<td>5.928</td>
</tr>
<tr>
<td>Coolant pump</td>
<td>0.0931</td>
<td>Coolant out</td>
<td>52.928</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
</tr>
</tbody>
</table>

Table 2.8: ORC energy balance

Table 2.8 shows the energy flows in and out of the entire system. The energy is calculated using air at 298 K and 1 bar as the reference state. The energy balance analysis yields a discrepancy of 10 W which when compared to the total energy flows in the system can be considered negligible. It can also be seen that 45.93 % of the exhaust energy has to be bypassed to maintain the design point. It is obvious that the exhaust energy is not being used optimally.

The cycle efficiency of the ORC alone while taking into account the parasitic losses from both pumps is given by:

\[
\eta_{ORC} = \frac{P_e - P_{fp} - P_{cp}}{\dot{Q}_{E2} - \dot{Q}_{E3}} = \frac{P_{net}}{\dot{Q}_{evap}}
\]  

(2.9.2)

The ORC efficiency obtained is consistent with literature findings [47,89,105,106].

The overall system efficiency provides an understanding of how much of the total
exhaust energy is being utilized. It is identical to Equation 2.9.3 but with use of $\dot{Q}_{E1}$ instead of $\dot{Q}_{evap}$ in the denominator.

$$\eta_{tot} = \frac{P_{net}}{\dot{Q}_{E1}}$$

(2.9.3)

Table 2.9: Exhaust energy breakup

<table>
<thead>
<tr>
<th>Component</th>
<th>%</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exh. bypassed</td>
<td>45.93</td>
</tr>
<tr>
<td>Absorbed into ORC</td>
<td>46.45</td>
</tr>
<tr>
<td>Evap. exh. out</td>
<td>7.58</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
</tr>
</tbody>
</table>

The Sankey diagram (Figure 2.20) provides a visual representation of the energy flows through the ORC. The magnitude of energy rejected through the bypass and coolant is evident here.

![Sankey diagram](image)

Figure 2.20: Energy flow through the ORC at design point
2.10 Operation of ORC system at Off-Design Conditions

Exhaust conditions fluctuate with variation in vehicle operating conditions. Hence, it is logical that the effect of differing exhaust conditions on the ORC should be studied. In this section, the performance of the ORC will be analyzed for a range of vehicle speeds when the transmission is in 8th gear assuming the vehicle is cruising on the highway.

In Figure 2.21, the red line represents the flow rate of the exhaust gases through the evaporator that is required to maintain the design point and hence also represents the minimum flow rate that has to be supplied by the engine. Following the red line, the exhaust mass flow rate required to maintain the design point lessens with increase in exhaust temperature.

The exhaust gas temperature and mass flow rate for a range of vehicle speeds are also plotted in Figure 2.21. The exhaust temperature and mass flow rate increases with increase in vehicle speed. At low vehicle speed, the exhaust flow rate is below the red line and hence there is not enough heat to maintain ORC design point. The
set of vehicle speeds where the exhaust flow rates are inadequate are indicated in magenta. This region below the red line can be referred to as the “Heat Limited Zone”. The exhaust conditions go above the red line at \(~53\) mph. In this region, there is an excess amount of exhaust and the design point is maintained by diverting this excess air away from the evaporator using the bypass valve. Let this region be referred to as the “Bypass Active Zone”. The vehicle speeds that are in the bypass active zone are represented by the blue line.

Figure 2.22 displays various system parameters graphed against vehicle speed. Here it can be seen that when vehicle increases speed from the design point which is in the bypass active region, the percentage of exhaust bypassed keeps increasing as both the exhaust mass flow from the engine and the temperature increase simultaneously and when reducing vehicle speed, percentage of exhaust bypass is reduced since the exhaust flow rate and temperature from the engine are reduced. Notice that the other two control inputs \(N_e\) and \(N_{fp}\) are constant. This behavior allows us to maintain the ORC by regulating the exhaust flow rate alone but only if the ORC is in the bypass active zone. Once the bypass reaches 0\%, i.e. all exhaust flow is directed into the evaporator, the ORC enters the heat limited zone and also loses use of the exhaust bypass valve as a control input. Here the refrigerant mass flow rate \(\dot{m}_{\text{ref}}\) has to be lowered by bringing down feed pump speed \(N_{fp}\) to reduce the heat requirements of the system in order to maintain the design point evaporator outlet temperature \(T_3\) and pressure \(p_H\). Reduction in \(N_{fp}\) without reducing expander speed \(N_e\) would cause the evaporator pressure \(p_H\) to diminish and therefore \(N_e\) is also reduced as seen in Figure 2.22.

In the bypass active region, \(Q_{\text{evap}}\), \(P_e\) and \(\eta_{\text{ORC}}\) stay constant since the ORC cycle is undisturbed with the bypass valve \(x_{BP}\) regulates the exhaust flow into the ORC. The total system efficiency \(\eta_{\text{tot}}\) alone changes since the heat input into the system
is the only variable here. $\eta_{ORC}$ decreases with increase in vehicle speed because the total heat energy supplied by the exhaust increases but the heat required by the ORC $Q_{evap}$ stays constant. While in the heat limited zone $\eta_{ORC}$ is relatively constant since the expander power output $P_e$ and heat input into the evaporator $Q_{evap}$ increase at roughly the same rate. Table 2.10 lists out relevant parameters of the ORC at different vehicle speeds with $P_e$, $\eta_{ORC}$ and $\eta_{tot}$ normalized with respect to the design point values.

Figure 2.22: System parameters vs vehicle speed
Table 2.10: Relevant Parameters of the Vehicle and ORC at Different Vehicle Operating Points

<table>
<thead>
<tr>
<th>Vehicle Parameters</th>
<th>Exhaust</th>
<th>Control Inputs</th>
<th>Targeted variables</th>
</tr>
</thead>
<tbody>
<tr>
<td>$V_{veh}$</td>
<td>$N_{eng}$</td>
<td>$\tau_{eng}$</td>
<td>$T_{E1}$</td>
</tr>
<tr>
<td>mph</td>
<td>RPM</td>
<td>N-m</td>
<td>K</td>
</tr>
<tr>
<td>45</td>
<td>1196</td>
<td>103</td>
<td>796</td>
</tr>
<tr>
<td>48</td>
<td>1261</td>
<td>111</td>
<td>807</td>
</tr>
<tr>
<td>50</td>
<td>1309</td>
<td>117</td>
<td>815</td>
</tr>
<tr>
<td>52</td>
<td>1362</td>
<td>124</td>
<td>825</td>
</tr>
<tr>
<td>53</td>
<td>1388</td>
<td>127</td>
<td>831</td>
</tr>
<tr>
<td>55</td>
<td>1440</td>
<td>134</td>
<td>846</td>
</tr>
<tr>
<td>60</td>
<td>1571</td>
<td>151</td>
<td>894</td>
</tr>
<tr>
<td>65</td>
<td>1702</td>
<td>170</td>
<td>924</td>
</tr>
<tr>
<td>70</td>
<td>1833</td>
<td>190</td>
<td>960</td>
</tr>
<tr>
<td>75</td>
<td>1964</td>
<td>211</td>
<td>990</td>
</tr>
<tr>
<td>80</td>
<td>2095</td>
<td>234</td>
<td>1023</td>
</tr>
<tr>
<td>85</td>
<td>2226</td>
<td>258</td>
<td>1065</td>
</tr>
<tr>
<td>90</td>
<td>2357</td>
<td>283</td>
<td>1106</td>
</tr>
</tbody>
</table>
2.11 Feed-Forward Control

Since the exhaust conditions are nowhere near constant under normal operation of a vehicle, if it is desired to maintain the ORC at the design point then a control system has to be designed to instruct the control inputs of the ORC to compensate for this unsteady exhaust flow. This was achieved with the use of a feed-forward controller. The feed-forward controller requires two inputs ($\dot{m}_{exh}$ and $T_{exh}$) that feed into three maps, one for each of the control inputs. The output of the feed-forward controller provides control input position that will maintain the design point. A schematic of the feed-forward control is displayed in Figure 2.23 and the maps used in the controller are shown in Figure 2.24.

In the bypass active zone, $N_{fp}$ and $N_e$ are held constant at their design point values while $x_{BP}$ varies. The bypass position values for the entire bypass active zone was generated as follows:

From the analysis in Section 2.10, it is known that the ORC operating point can be kept unchanged for varying exhaust conditions by simply varying the exhaust bypass $x_{BP}$ in the bypass active zone. Hence, $\dot{m}_{E2}$ required to maintain the design point for exhaust gas temperatures between $573^\circ C$ and $833^\circ C$ can be simply done by adjusting the exhaust flowing into the evaporator with the help of a PI controller. The PI controller can use the error signal of either $T_4$ or $p_H$ to determine $\dot{m}_{E2}$. During this process, $N_{fp}$ and $N_e$ are kept constant.
Once $\dot{m}_{E2}$ for the exhaust temperature range was determined, it was then used to generate $x_{BP}$ values for $\dot{m}_{exh}$ between 0.022 kg/s and 0.07 kg/s using the formula $x_{BP} = 1 - \frac{\dot{m}_{E2}}{\dot{m}_{E1}}$. In the heat limited zone, unfortunately no such shortcut or any automation could be used and the control inputs had to be manually tuned for a many points until the feed-forward map was adequately populated.

The performance of the ORC system with the feed-forward map was then validated with the GT-POWER model. The step response to three disturbances were tested:

a) 10% increase in exhaust mass flow rate $\dot{m}_{E1}$ ($T = 1000$ s to $1500$ s)
b) 10% increase in exhaust temperature $T_{E1}$ ($T=2000\text{ s to }2500\text{ s}$)

c) $\dot{m}_{E1}$ and $T_{E1}$ reduced to their values at 50 mph. ($T=3000\text{ s to }4500\text{ s}$)

The results of the validation are in Figure 2.25.

Figure 2.25: ORC feed-forward map performance

When the exhaust gas flow rate is increased at $T=1000\text{ s}$, the exhaust bypass opens up to keep the flow rate through the evaporator constant. If this compensation is enough to keep $\dot{m}_{E2}$ constant then no other transients will occur in the ORC which is what happens as seen in Figure 2.25. When the exhaust temperature is increased though (at $T=2000\text{ s}$), $\dot{m}_{E2}$ has to be reduced by increasing exhaust bypass to keep heat transfer into the evaporator $\dot{Q}_{evap}$ constant. A slight transient is observed in $p_H$ and $T_4$ when the exhaust temperature step is applied and removed. While the previous two disturbances were within the bypass active zone at all times, the performance of the feed-forward controller at the heat limited zone was checked by
taking $\dot{m}_{exh}$ and $T_{exh}$ to levels corresponding to 50 mph at $T=3000$ s. Here, all the exhaust is routed through the evaporator and both $N_{fp}$ and $N_e$ are reduced to make up for the reduced heat load of the exhaust. $p_H$ seems to settle 1 bar below the design point specifications and there is also $\sim 5$ K decrease in $T_4$. This error can be attributed to interpolation inaccuracies in the maps and errors in tuning.
CHAPTER 3
INTRODUCTION TO PARTICLE SWARM OPTIMIZATION

Mathematical optimization involves seeking the best combination of inputs to minimize (or maximize) an objective function. An optimization problem (OP) with a single objective function can be expressed as follows:

\[
\min_{x \in \Omega} f(x) \tag{3.0.1a}
\]

\[
h_i(x) = 0, \quad i = 1, 2, ..., p \tag{3.0.1b}
\]

\[
g_i(x) \leq 0, \quad i = 1, 2, ..., q \tag{3.0.1c}
\]

where \( x = [x_1, x_2, ... x_m] \) and \( \Omega \in \mathbb{R}^m \) is the feasible set of variables. The functions \( h_i(x) \) and \( g_i(x) \) are equality and inequality constraints on the variables of the optimization problem respectively.

Before we delve into the Particle Swarm Optimization (PSO) algorithm, a brief introduction to numerical optimization in general will be provided.

3.1 Classification of Optimization Problems

An optimization problem can be broadly classified under the following categories [16]:

1. Constrained and Unconstrained optimization;
2. Global and Local optimization;

3. Static and Dynamic optimization.

**Constrained and Unconstrained Optimization**  Optimization problems with the formulation described in Equation 3.0.1 that have constraints \( h_i(x) \) and \( g_i(x) \) are called constrained optimization problems and if no such constraints exist (i.e. \( p = q = 0 \)) then they are termed unconstrained optimization problems with the latter being much simpler to solve [16]. Many constrained optimization problems can be simplified to unconstrained problems in cases where the bounds on the individual elements of vector \( x \) or inequality constraints that are known not to be in the region of the optimum since they do not affect the final result. Equality constrained problems are generally solved using method of Lagrange multipliers and if inequality constraints are present then the Karush-Kuhn-Tucker (KKT) equations, a generalized form of the method of Lagrange multipliers are made use of [107].

**Global and Local Optimization**  Non-Linear functions may have multiple local solutions in existence and solving such problems becomes difficult for many traditional optimization algorithms that are based on gradient based methods [16, 108]. These methods have a tendency to converge to a local solution and the solution is highly dependent on the starting point of the algorithm [16]. However, local optimization methods are used in global optimization algorithms such as branch and bound method to arrive at a global optimum [109]. Metaheuristic algorithms are usually employed to solve such problems [110]. Metaheuristic algorithms generally have a stochastic underpinning that enables efficient search of a large set of feasible solutions [108].

**Static and Dynamic Optimization**  When the objective function is time varying and has to be optimized over a period of time it falls under Dynamic Optimization,
also referred to as optimal control in control theory [111]. An example of dynamic optimization would be reducing control effort going into a system over finite or infinite time. There are three approaches to tackling dynamic optimization problems, the classical method is the extension of calculus of variations to the continuously time varying paradigm, upon which optimal control is based [111]. Dynamic Programming introduced by Richard Bellman in the 1950’s approaches the dynamic optimization in discrete time and is therefore very convenient for implementation in computers and final approach is using the Lagrangian method [112, ch.7].

### 3.2 Solutions to Optimization Problems

Multiple methods exist to solve these problems depending on the nature of the objective function and its constraints. Solution methods may be divided into the following categories:

1. Brute force
2. Calculus of variation
3. Iterative approaches
4. Metaheuristic algorithms

The simplest of all the above methods is by brute force, i.e. exhaustive evaluation of the entire feasible search space. This is by far the most inefficient method of solving an optimization problem. Calculus of variation is suitable for situations where the objective function has an analytical formulation. This method cannot be used in cases where the objective function is highly non-linear or simply unavailable. The iterative approach is the more popular optimization method due to its suitability for computer implementation [113,114]. This method will be discussed a little more in detail later.
in this section due to the ubiquity of its use in general engineering applications and the relevance to this thesis [113, 114].

3.2.1 Overview of Iterative Optimization Methods for Static Non-Linear Problems

The approach to solving a non-linear problems using iterative techniques depends on its formulation. If the problem has a linear objective function and linear constraints it falls under the Linear Programming category, where the simplex algorithm and more recently Interior-Point methods are common Linear Programming methods to find the optimum [16]. In the case of non-linear objectives with and without constraints, the optimization problem has to be solved with Non-Linear Programming (NLP) methods such as Quadratic Programming, Convex Programming, etc. The type of NLP method to use depends on the nature of the objective function and constraints.

The general step update equation shown below forms the heart of most iterative optimization algorithms that use gradient information of the objective function as a part of the algorithm.

\[ x_{k+1} = x_k + \alpha_k p_k \]  

(3.2.1)

where \( p_k \) is the step direction and \( \alpha_k \) the step length. Gradient based algorithms can be subdivided into two further categories based on how the second term of the step update is determined, namely, Line Search methods and Trust Region methods. In Line Search methods, \( \alpha_k \) and \( p_k \) are updated in separate steps for every iteration while the Trust Region method does both these operations in a single step. The formulation for the step update equation \( p_k \) for a few common Line Search methods is shown in Table 3.1 [16]. For a more exhaustive description of these methods, the reader is referred to [16].
Table 3.1: Step direction update equations [16]

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>Update Equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Gradient Descent</td>
<td>( p_k = \nabla f_k )</td>
</tr>
<tr>
<td>Conjugate Gradient</td>
<td>( p_k = -\nabla f_k + \left( \frac{| \nabla f_k |}{| \nabla f_{k-1} |} \right)^2 p_{k-1} )</td>
</tr>
<tr>
<td>Newton’s method</td>
<td>( p_k = \frac{\nabla f}{\nabla^2 f} )</td>
</tr>
</tbody>
</table>

Constrained Optimization Techniques

The KKT equations currently form the basis of most constrained optimization methods currently in use [16]. As mentioned before, the KKT equations are a generalized form of the method of Lagrange multipliers. The KKT condition below is an add-on to the method of Lagrange multipliers to accommodate inequality constraints.

\[
F(x, \lambda, \mu) = f(x) + \sum_{i=1}^{p} \lambda_i h_i(x) + \sum_{i=1}^{q} \mu_i g_i(x) \quad (3.2.2a)
\]

\[
h_i(x) = 0, i = 1, 2, ..., p \quad (3.2.2b)
\]

\[
g_i(x) \leq 0, i = 1, 2, ..., q \quad (3.2.2c)
\]

\[
\mu_i \geq 0, i = 1, 2, ..., q \quad (3.2.2d)
\]

where \( \lambda \) and \( \mu \) are referred to as KKT multipliers. The solution to the KKT equations is a stationary point which is a necessary condition for a point to be an extrema. This is incorporated into the iterative methods to solve constrained optimization problems.

The most effective method currently available for solving non-linear constrained optimization problems is Sequential Quadratic Programming (SQP) [16]. SQP works
by solving a Quadratic Programming (QP) subproblem at every iteration. QP methods are used to solve quadratic objective functions with linear equality constraints. In SQP the objective function is converted into a QP by taking the Taylor’s series quadratic approximate of the objective function and linearizing the equality constraints (see Equation 3.2.3) at the iterate $x_k$. This QP subproblem is solved for every iteration of the SQP until the minimum is reached.

**Quadratic subproblem:**

$$
\min_d \frac{1}{2} d^T \nabla^2 f(x_k) d + \nabla f(x_k)^T d \quad (3.2.3a)
$$

$$
\nabla h_i(x_k) + \nabla h_i(x_k)^T d = 0, \quad i = 1, 2, ..., p \quad (3.2.3b)
$$

$$
\nabla g_i(x_k) + \nabla g_i(x_k)^T d \leq 0, \quad i = 1, 2, ..., q \quad (3.2.3c)
$$

where $d = x_{k+1} - x_k$. A Quasi-Newton update mechanism such as SR1, BFGS or DFP [16] can be utilized for updating the Hessian matrix ($\nabla^2 f(x_k)$) and either a Line Search or Trust-Region algorithms can be made use of to perform step updates.

The SQP is adapted for equality constrained optimization by replacing $f(x_k)$ with its Lagrangian $\mathcal{L}(x_k, \lambda_k)$.

Inequality constrained optimization problems use the “active-set” algorithm. Here, the algorithm maintains an active set of constraints and the inequality constraints are added or removed from this active set depending on whether they are active at each iteration and only this active set of constraints is used to solve the QP. The performance of the PSO will be benchmarked against the active-set algorithm later in the chapter.

For a more exhaustive explanation of the active-set algorithm’s implementation, refer to the Optimization Toolbox section of MATLAB® help and also refer to [16, ch. 16] for further reading on the active set algorithm and gradient based optimization algorithms in general.
3.2.2 Metaheuristic Algorithms

Metaheuristic algorithms are a branch of optimization algorithms that do not require any information of the system to arrive at a solution. They usually have a high level algorithm that guides the search towards the optimum based on the behavior of the system with respect to changes in input variables. The Particle Swarm Optimization algorithm which will be used in this thesis falls under this broad category. Other algorithms such as Ant Colony Optimization, Simulated Annealing, Tabu Search and Evolutionary Algorithms come under this class [108]. All of these algorithms are not deterministic since they have a stochastic element to them that aids in their effectiveness. Advantages of these class of algorithms include capability of working with highly discontinuous objective functions and their ability to escape from local optima aided by the stochastic parameters embedded in the algorithm [108]. Many metaheuristic algorithms are also population based in which a set of solutions are evaluated concurrently. The Particle Swarm Optimization algorithm, Evolutionary Algorithms and Ant Colony Optimization work under this mechanism while Tabu Search and Simulated Annealing are single solution algorithms [108,110]. The Particle Swarm Optimization algorithm will be discussed in more detail in the next section. [108] goes more into detail about Metaheuristic algorithms.

3.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is a stochastic, population based global optimization algorithm, devised to optimize nonlinear functions. The algorithm was invented by Kennedy and Eberhart in 1995 [115], inspired by the behavior of a flock of bird or a school of fish in nature foraging for food. The inventors of the algorithm, in [115] quoted the following paragraph by the sociobiologist E.O. Wilson in
his then controversial book *Sociobiology* [116] to provide insight into the working of the algorithm:

_In theory at least, individual members of the school can profit from the discoveries and previous experience of all other members of the school during the search for food. This advantage can become decisive, outweighing the disadvantages of competition for food items, whenever the resource is unpredictably distributed in patches._

Population based algorithms also include genetic algorithms [117], genetic programming [118], evolutionary strategies [119] and evolutionary programming [120]. The aforementioned algorithms are all based on the concept of biological evolution, hence the umbrella term Evolutionary Algorithms. The PSO differs from these in that its origins are based on the sociological aspect of nature.

### 3.3.1 Algorithm Description

The PSO algorithm controls the motion of a swarm of particles within the search space with each particle representing a potential solution to the optimization problem.

Consider the swarm size to be $s$. The population can be initialized randomly within the search space using a pseudo-random generator. The initial seed swarm can also be uniformly distributed over the search space using a quasi-random sequence such as Sobol sequence or a Latin hypercube distribution [121, p. 23]. The basis of the PSO algorithm are the velocity and position update equations below:

$$v_i(t + 1) = \omega v_i(t) + r_1 c_1(pbest_i - x_i(t)) + r_1 c_1(gbest - x_i(t)) \quad (3.3.1a)$$

$$x_i(t + 1) = x_i + v_i(t + 1) \quad (3.3.1b)$$
where

\[ v \quad \text{particle velocity vector, } v = [v_1, v_2, \ldots, v_m] \]

\[ x \quad \text{particle location vector, } x = [x_1, x_2, \ldots, x_m] \]

\[ \omega \quad \text{inertia weight} \]

\[ p_{best} \quad \text{particle personal best location} \]

\[ g_{best} \quad \text{global best of all particles} \]

\[ c_1 \quad \text{cognitive parameter} \]

\[ c_2 \quad \text{social parameter} \]

\[ r_1, r_2 \quad \text{stochastic factors in the range [0,1]} \]

\[ i \quad \text{particle number } i = 1, 2, \ldots, s \]

\[ t \quad \text{iteration or timestep} \]

The first term of the velocity equation (3.3.1a) is the \textbf{inertia term}. It stores the particles velocity in the previous iteration. This term tends to cause the particle to stray out of the general area of the swarm. The second and third part of the velocity equation (Equation 3.3.1a) are its \textbf{cognitive} and \textbf{social components} respectively. While the particle \( i \) traverses the search space, the cognitive component stores in \( p_{best_i} \) the best position held by the particle \( i \) until the present iteration. \( p_{best_i} \) serves as an attraction point for the particle. \( g_{best} \) is the best position considering the swarm in its entirety up until the present iteration \( t \) and serves as a second attraction point for all the particles in the swarm. \( p_{best_i} \) and \( g_{best} \) act as a “nostalgia” factor, attracting the particle back towards those points where it had the best “experience” with \( g_{best} \) serving as a communication port that links all particles together. Figure 3.1 graphically illustrates the movement of one particle in the search space to the next iteration.
The terms $c_1$ and $c_2$ are weighting factors (also called as acceleration coefficients) and can be adjusted to enhance the cognitive or social aspect of the swarm. The stochastic factors $r_1$ and $r_2$ are the source of randomness in algorithm. The inventors of the PSO set both the weighting factors to 2 with the aim that product of the weighting and stochastic factors average out the overshoot and undershoot of the particle with respect to $p_{best_i}$ and $g_{best}$. These three components are then used to calculate the velocity and then added to the current position of the particle to obtain the next position.

In the original version of the PSO, the velocity term was without the inertia weight [115]. [122] in 1998, added the inertia term to allow fine tuning the behavior of the PSO. The authors concluded that the best range for the inertia term would be $[0.9,1.2]$. The authors also observed that the inertia term allows further exploration of new locations in the search space preventing the swarm from prematurely converging to a local optimum. A higher inertia weight $\omega$ value while allowing the swarm to
explore search locations other than that of the neighborhood of initial seed, slows down convergence. A low inertia weight would restrict the swarm to its initial search space causing it to behave more like a local search algorithm. The authors also briefly mention linearly varying the inertia weight over the course of optimization process from an initially high value and settling on a smaller weight after a few iterations. Doing so would favor exploration of new search areas initially, then with decrease in inertia weight as the iterations increase, the swarm will start to exploit the search space better. This makes sure that the swarm adequately explores the entire search space before refining the search. This concept was further investigated by [123] in 1999. The authors linearly decreased the inertia weight from 0.9 to 0.4, concluding that this offered better performance than using a constant inertia weight.

In [124] the authors compared the performance of linearly decreasing inertia weights against a range of constant inertia weights. The linearly decreasing inertia weights was between 0.9 and 0.4 for the first 1500 iterations. The population size was 20 particles and Schaffer’s $f6$ [125] function was the test function used. The linearly decreasing weights performed better than the constant inertia weights consistently.

A velocity clamping term ($V_{max}$) was used by Kennedy [126] in 1997 to limit the maximum velocity of the particles, claiming to prevent the particle velocities from reaching too high a value, ultimately causing the swarm to diverge and also improving performance in the process too. The effect of inertia weight and the velocity clamping was further studied by Shi and Eberhart [124] in 1998 on Schaffer’s $f6$ [125] function. They observed that for $V_{max} \leq 2$, $\omega = 1$ performed best and for $V_{max} \geq 3$, $\omega = 0.8$ performed best. They concluded that setting maximum velocity to half the range $(\frac{1}{2}(Upperbound - Lowerbound))$ of the search space would be the safest choice if no knowledge of the right $V_{max}$ is known.
With regards to the effect of population size on PSO performance, a high population size may increase the probability of finding the global minimum, especially for an objective function with many local minima but also results in increase in function evaluations that inflates computation time which is a cause for concern in situations where there is a high cost involved in evaluating the objective function. On the other hand, using a small population size might lead to the swarm converging to a local minima. Hence, a balance must be struck between lower function evaluations and higher success rate. In [127], a study was performed to find guideline values for a variety of PSO parameters that can be suited for general use. The authors concluded that a population size of 30 provided a reasonable compromise.

Insofar, only empirical studies have been mentioned, a short summary of mathematical analyses on the PSO will now follow. One of the first published mathematical analysis on the PSO is by Ozcan and Mohan [128]. The authors found that the swarm follows a sinusoidal wave form and described this phenomenon as particles "surfing" on sine waves. They also conjectured that the velocity clamping helped a particle "jump" onto another wave. Later, Bergh [121, sec. 3] provided a rigorous mathematical analysis of the convergence of the PSO and derived conditions for PSO stability. Bergh also provides a mathematical reasons for the "surfing" phenomenon of the particles observed by Ozcan and Mohan [128] proving that the choice of \( c_1 = c_2 = 2 \) in the original PSO [115] caused the PSO to be marginally stable resulting in the sinusoidal behavior of the swarm requiring the use of velocity clamping. The conditions for guaranteed convergence of the PSO to an equilibrium point as derived by Bergh are shown below. The conditions were also derived in [11]:

\[
0 \leq \varphi \leq 4 \\
\frac{\varphi}{2} - 1 \leq \omega \leq 1
\]

where \( \varphi = c_1 + c_2 \).
Clerc and Kennedy [129] also performed a detailed mathematical analysis on the PSO algorithm and their work indicated that using a constriction factor on the velocity equation (see Equation 3.3.2) assured stabilization of the PSO without the need of using the velocity clamping term. The constriction factor method was used assuming \( \varphi = 4.1 \) yielding a constriction factor \( \chi = 0.7298 \) which is equivalent to \( \omega = 0.7298 \) and \( c_1 = c_2 = 1.4962 \) in the notation of the general form of the velocity equation as in Equation 3.3.1a. The aforementioned parameters for the PSO proved to be robust as tested in [130] over a range of test functions.

\[
v_i(t + 1) = \chi \left( v_i(t) + r_1 c_1 (pbest_i - x_i(t)) + r_1 c_1 (gbest - x_i(t)) \right)
\]  

(3.3.2)

where

\[
\chi = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}
\]

(3.3.3)

and \( \varphi = c_1 + c_2, \quad \varphi > 4 \)

Eberhart and Shi [130] confirmed that the constriction method performed better than the usual approach using 5 test functions. They also observed that the particles in the constriction method had a slight tendency to spread out into unnecessary areas of the search space. They found that clamping the maximum velocity of the constriction method to \( V_{max} \) improved results. Similar results were obtained by Carlisle and Dozier [127].

Finally, Trelea [131] discussed a simplified mathematical analysis of the PSO, building on top of Clerc and Kennedy’s [129] work. Tralea characterized the behavior of the PSO over combinations of PSO inertia weight and cognitive parameters and demarcated regions where the PSO was damped, overdamped, underdamped and marginally stable based on these parameters.

It must be mentioned that all the above mentioned mathematical analyses simplified the PSO by ignoring the stochastic parameters, thereby converting the PSO to
a deterministic system. Also $pbest$ and $gbest$ are assumed constant resulting in the particles becoming independent of one another.

### 3.3.1.1 Constraint Handling

The PSO algorithm does not have a constraint handling mechanism built into it. Constraint handling methods used in Genetic Algorithms were usually adopted. [17] provides a detailed review of constraint handling techniques used in Evolutionary Algorithms (EA’s) over the years. A brief overview of the most common and simplest techniques will be given below.

The oldest constraint method used in optimization algorithms was the penalty function method [17]. Here, a penalty term is added to the objective function based on the degree to which it violates a constraint. Penalty functions can be classified as interior and exterior. Exterior penalty functions guide the particle from an infeasible region to the feasible region by reducing the penalty as the particle moves closer to the feasible region while interior methods cause the particle’s objective function in the feasible region tend to infinity as it approaches the feasible region boundary. Exterior methods are used in EA’s because infeasible points are allowed to exist at initialization. The general form of the exterior penalty function method is:

$$\tilde{f}(x) = f(x) \pm \left[ \sum_{i=1}^{p} r_i \times G_i + \sum_{j=1}^{q} c_i \times H_j \right]$$

(3.3.4)

where $\tilde{f}(x)$ is the new modified objective function, $G_i$ and $H_i$ are functions of the equality $g(x)$ and inequality $h(x)$ constraints respectively. $r_i$ and $c_i$ are termed "penalty factors". A major disadvantage of this method is that it is considerably difficult to choose the right penalty factor and it varies with the optimization problem. The penalty factor can either be constant or a function of other variables such as the iteration number. The simplest penalty method would be the Death penalty
where the offending particle is simply rejected from consideration as an optimum point. [17] provides a detailed review of all the types of penalty function methods used in EA’s.

Figure 3.2: Violated design point redirection. (a) Particle outside feasible region. (b) Particle position redirection. [11]

[132] suggested ignoring the inertia term in the PSO velocity equation (3.3.1a) when a particle strays out of the feasible region so that it can be steered back into the feasible region more effectively. Figure 3.2 illustrates this concept [11]. This method is implemented in addition to the penalty function method [11, 133].

In the illustration, the subscript refers to the iteration number and the superscript refers to the particle number, which is flipped from the convention followed in this thesis.
The PSO algorithm also does not have an inbuilt method to restrict the movement of the particles within the search space. Restricting movement of particles within a boundary might be useful in optimizing systems where values outside the boundary might not be physically possible. [12] was the first to introduce the concept of boundary limits. The authors proposed three boundary conditions, namely, Absorbing walls, Reflecting walls and Invisible walls. The description is reproduced verbatim below [12] with illustration of the concept in Figure 3.3,

1. Absorbing Walls: When a particle hits the boundary of the solution space in one of the dimensions, the velocity in that dimension is zeroed, and the particle will eventually be pulled back toward the allowed solution space. In this sense the boundary walls absorb the energy of particles trying to escape the solution space.

2. Reflecting Walls: When a particle hits the boundary in one of the dimensions, the sign of the velocity in that dimension is changed and the particle is reflected back toward the solution space.

3. Invisible Walls: The particles are allowed to fly without any physical restriction. However, particles that roam outside the allowed solution space are not evaluated for fitness. For nearly all engineering
applications, the computationally expensive portion of the algorithm is the fitness evaluation. The motivation behind this technique is to save computation time by only evaluating what is in the allowed solution space, while not interfering with the natural motion of the swarm.

The authors reported that all three methods produced similar results when tested with the Rosenbrock [134] and Griewank [135] functions. The exception being the invisible walls boundary condition on the Griewank function where it outperformed the other two methods. It must be noted that these boundary condition handling techniques do not restrain the particle from straying outside the search space but only modify the velocity after the fact so that its effect is shown in the inertia weight of the next iteration. In [136] where the PSO is used in antenna design optimization, the author proposed two terms: hard and soft boundary conditions. The soft boundary condition is the same as previously outlined. The hard boundary condition, in addition to the velocity clipping criteria, also restricts the physical movement of the particle within the boundary of the search space. The author reports that the hard boundary condition provided better results. The author also observed their particular optimization problem was sensitive to $V_{\text{max}}$ and hence mentions that $V_{\text{max}}$ must also be carefully chosen. [137] proposed the damping boundary which is a hybrid between the absorbing and reflecting boundaries. Here, not only is the sign of the velocity flipped as in the reflecting wall method but also a multiplier is applied in the form of a random number in the range of [0,1]. This means that the particle may not only be completely reflected or absorbed but also anything in between the two. It must be noted that the author does not mention restricting the position of the particle and hence it is assumed that the author used a soft boundary condition as previously
mentioned and defined by [136]. The author reported that the damping boundary provided results consistently better or equivalent to the other methods.

3.3.2 Description of Sample Problem

The capabilities of the PSO algorithm will be benchmarked against a few other optimization methods. This benchmarking will be done with the Rosenbrock function [134] shown in Equation 3.3.5. The Rosenbrock function has a long parabolic shaped valley in which the global minimum \( (x^*_w) \) is present at \( f(1,1) = 0 \). The inequality constraint shown in Equation 3.3.6 is imposed, which is a circle with center (1,1) and unit radius. Hence, all locations within the circle including the global minimum will be infeasible points. The new minimum \( (x^*_c) \) at \( f(1.3860, 1.9224) = 0.1492 \) is situated on the boundary of the constraint. Due to the constraint imposed, there is also a local minimum \( (z) \) at \( f(0.4281, 0.1796) = 0.3284 \). Depending on the starting point, the optimization solvers will have a tendency to converge to this location. The search space was limited to \([-1.5, 2]\) and \([-0.5, 3]\) for the x and y axes respectively.

\[
x^* = \arg \min_x f(x)
\]

\[
f(x) = \sum_{i=1}^{n-1} \left[ 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right]
\]  

\[
s.t \quad (x - 1)^2 + (y - 1)^2 \geq 0
\]  

94
The PSO was benchmarked with the active-set algorithm and an exhaustive manual grid search with the latter being the most crude and simplest optimization method. The active-set algorithm is present in the MATLAB® Optimization Toolbox.

3.3.3 Application of PSO to Sample Problem, Results and Comparison with Other Optimization Methods

The PSO setting used for the benchmarking are displayed in Table 3.2. The acceleration coefficients as recommended by [127] were used. A linearly decreasing inertia weight was implemented with the starting and ending points as recommended by [123,124]. A hard boundary [136] was used to prevent particles from straying out of the search space. To handle constraints, the inertia term was zeroed out when a constraint was violated [11,132,133] and the particle taken out of contention for $p_{best}$.


<table>
<thead>
<tr>
<th>Property</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population Size</td>
<td>50</td>
</tr>
<tr>
<td>$V_{max}$</td>
<td>$0.8 \times 2X_{max}$</td>
</tr>
<tr>
<td>$c_1$ &amp; $c_2$</td>
<td>2.8 &amp; 1.3</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.9 to 0.4 over 150 generations</td>
</tr>
</tbody>
</table>

and $g_{best}$. The starting point for the active-set algorithm was randomized for every trial.

The benchmarking results are averaged over 20 trials. The results of the benchmarking are in Table 3.3. From the table it is clear that the PSO is more successful in tracking the global minimum than the active-set algorithm. The active-set algorithm consistently converged in the least number of iterations but has a higher computation time. This suggests higher overhead processing costs. If the cost of evaluating a function is significantly higher than the overhead of the active-set algorithm, like for example using a complex Simulink® model instead of just calculating a simple algebraic equation, then the PSO may require higher computational time due to the higher number of function evaluations but this effect can be mitigated in a parallel processing environment where each particle can be independently evaluated in a separate core.
### 3.3.4 Implementation of PSO for Multiobjective Optimization

A Multi-Objective Optimization Problem (MOP) involves simultaneously optimizing more than one objective. The mathematical formulation can be stated as follows:

\[
\min_{x \in \Omega} \ (f_1(x) \ f_2(x) \ \ldots \ f_n(x))
\]  

(3.3.7a)

\[
h_i(x) = 0, \quad i = 1, 2, \ldots, p
\]  

(3.3.7b)

\[
g_i(x) \leq 0, \quad i = 1, 2, \ldots, q
\]  

(3.3.7c)

where \(x = [x_1, \ x_2, \ \ldots \ x_m]\) and \(\Omega\) is the feasible set of variables. In an MOP, there may not be one single “optimum” solution but a family of such solutions where one objective cannot be improved without resulting in the deterioration of at least one other objective function. Such solutions are called Pareto Optimal solutions. This is called the concept of Pareto Optimality. Formally it can be defined as follows:

**Definition 1.** Considering there are \(n\) objectives to be optimized (henceforth assuming minimization), the vector \(x^*\) is Pareto Optimal(or Pareto efficient) if and only if:

\[
\forall i \ f_i(x^*) \leq f_i(x) \quad \land \quad \exists i \ f_i(x^*) < f_i(x), \ i = 1, 2, \ldots, n.
\]
where \( f(x) = \{ f_1(x), f_2(x), ..., f_n(x) \} \) is the objective vector and \( x \) represents all possible solutions in the search space. The corresponding objective function \( f(x^*) \) is called a non-dominated point. A set of all such solutions is called the Pareto Optimal set in the search space and non-dominated set in the objective space.

The terms Pareto Optimal Set and non-dominated set have tended to be used interchangeably though not technically correct. The non-dominated set is also called as the Pareto Front. It should be stated that there can be an infinite number of solutions in the Pareto Front. The aim of a Multi-Objective Optimization (MOO) algorithm is to approximate the Pareto Front. Another term used in MOO is Pareto Dominance defined as follows:

**Definition 2.** \( x^* \) is be said to dominate \( x' \) if and only if:

\[
    f_i(x^*) \leq f_i(x') \quad \land \quad \exists_i f_i(x^*) < f_i(x'), \; i = 1, 2, ..., n.
\]

This is denoted as \( x^* \preceq x' \).

Evolutionary Algorithms (EA) are well suited for MOO since the EA can explore multiple Pareto Efficient candidates solutions in a single execution of the algorithm as opposed to running a gradient-based algorithm multiple times [139]. It is also easier to adapt a EA to the MOO environment.

The simplest method of using any optimization algorithm to generate Pareto Optimal points is the weighted sum method. In this method the objective vector is condensed to a single objective function through a weighted combination of all its elements as shown below:

\[
    \tilde{f}(x) = \sum_{i=1}^{n} \alpha_i f_i(x) \quad \land \quad \sum_{i=1}^{n} \alpha_i = 1 \quad (3.3.8)
\]

By running the optimizer for a range of weights, the Pareto Front can be constructed. It must be noted that the minimum of a every unique combination of
weights is necessarily not a Pareto Optimal Point and hence might cause wasted computational time [140]. Also, the weighted sum method does not produce an even distribution of points on the Pareto front for a corresponding even distribution of a weight combinations in every case and is very dependent on the shape of the Pareto Front [141]. Another issue with this method is that if the Pareto Front has non-convex regions then those points that lie on this region are unattainable with the weighted sum method [141–143]. [144] provides a survey of other objective function aggregation approaches.

An early example of using the weighted sum method would be [145]. Here the authors used the Generalized Reduced Gradient (GRG) algorithm in combination with weighted sum method to optimize a thermal distribution piping network system.

The first Multi-Objective implementation of any EA is the Vector Evaluated Genetic Algorithm (VEGA) [146]. The algorithm works by dividing the population into sub-population and each sub-population optimizes for a single objective function and then shuffling between the sub-populations occurs, after which crossover and mutation operators at every iteration to maintain population diversity. The algorithm is based on the concept that the optimum of any single objective will be a vertex of the Pareto optimal set. This drawback of this method is that it does not produce an effective distribution of points on the Pareto front but the inventor of the algorithm suggested providing all non-dominated points a ”heuristic selection preference” during the crossover operations to partly rectify this. [139] lists many other methods that do not explicitly use the concept of Pareto Optimality.

[117] was first to propose using Pareto Optimality as a fitness factor in Genetic Algorithms (GA) to improve upon the VEGA algorithm’s non-dominant point clustering problem, ushering in a new class of GA’s called Pareto Dominance based approaches. The author suggested assigning importance to every point as follows: All
non-dominated points of the current generation are assigned rank 1 and temporar-
ily removed from the population, then the non-dominated points from the remain-
ing population are assigned rank 2 and the process is continued till all members of
the population have been ranked. The fittest particle being the lowest rank. The
NSGA [147] and NSGA-II [148] algorithms are based on this ranking method. [149]
modified [117]’s method by ranking each member of the population based on the num-
ero of individuals that it is dominated by. Refer to [150] for a more thorough study
on MOO. The advantages of Pareto based approaches over the weighted aggregate
method is that they provide a more even spread of points over the Pareto Front and
are less sensitive to the shape of the Pareto Front.

The first documented instance of the PSO being used for MOO was reported
in [151]. Here, the authors proposed storing the set of non-dominated solutions
found by an individual particle in repositories that are constantly updated in every
iteration. The individual best for every iteration is chosen from this repository at
random. The social best for the particle is chosen by selecting a non-dominated
solution within its neighborhood of particles. The authors reported promising results
with this algorithm.

In 2002, the Multi-Objective Particle Swarm Optimization (MOPSO) [17] al-
gorithm was one of the first published MOO PSO algorithms among others such
as [152, 153]. The MOPSO is based on the Pareto Dominance approach. The al-
gorithm makes use of a continuously updated repository to store all non-dominated
solutions obtained so far and an adaptive grid crowding distance strategy inspired
by the Pareto Archived Evolution Strategy (PAES) algorithm [154]. The gbest for
every iteration is chosen from the repository through a random weighted probability
method based on the crowding of the non-dominated solutions. The MOPSO is a
relatively simple but effective algorithm hence it’s use in the optimization of the ORC

100
later in Chapter 4. A detailed explanation of the MOPSO algorithm will be provided in the next section.

The reader is pointed towards the following for a more detailed survey of MOO methods for further reading. [155] provides a brief review of MOO methods while [156] reviews EA based MOO methods in detail providing their advantages and disadvantages and [144,157] review MOO implementations of PSO.

3.3.4.1 MOPSO algorithm

The MOPSO algorithm as described in [17] is posted in Algorithm 1. Consider the size of the swarm to be $S$, position vector matrix $POS$ with dimension $s \times m$, velocity vector matrix $VEL$ with dimension $s \times m$. The seed swarm is first initialized within the search space with velocity set to 0. The objective functions for all the particles are then evaluated. The particles are then checked for non-domination and the non-dominated particles’ positions in the search and objective space is stored in the repository $REP$. The $gbest$ for the every iteration is chosen from this repository based on roulette wheel selection, a random weighted probability selection. This is to encourage even spreading of the Pareto Front by making sure the non-dominated point that is furthest away from the rest of the non-dominated points has a higher probability of getting chosen as $gbest$. Probability weights are assigned based on fitness assigned as follows: The entire objective space is divided into hypercubes and each hypercube that has a particle from the repository present in it is assigned a fitness $fit = 10/x$ where $x$ is the number of particles within the hypercube. A hypercube is selected based on the roulette wheel selection and a particle within the chosen hypercube is selected at random to be the $gbest$ for that iteration. The $pbest$ is initialized to be the initial position.

The algorithm now enters the iterative loop. The $gbest$ for the iteration is selected
Algorithm 1 MOPSO

1: Initialize swarm position POS
2: Initialize VEL = 0
3: Evaluate all particles
4: Find non-dominated solutions and add their positions into repository REP
5: Classify particles in REP into hypercubes based on their position in the objective function space.
6: Initialize PBEST = POS
7: while ITER < MAXITER do
8: Select $REP[h]$ for the current iteration.
9: $VEL = W \times VEL + R_1 \times (PBEST - POS) + R_1 \times (REP[h] - POS)$
10: POS = POS + VEL
11: Restrict particle position within the search space.
12: Evaluate all particles.
13: Update repository and hypercube. If repository is full then remove particles at random starting with the least populated hypercube.
14: if POS dominates PBEST then
15: PBEST = POS
16: else if PBEST dominates POS then
17: no change in PBEST
18: else neither dominates
19: pick either one at random as the new PBEST.
20: end if
21: ITER = ITER + 1
22: end while
by the roulette wheel method and the new velocity and position for all the particles is calculated. The new position is forced to be within the search space boundary. The particles are evaluated at their new positions and the repository and hypercube are updated accordingly along with the fitness of each hypercube. The size of the repository is limited to a finite value since filtering the non-dominated points is a computationally intensive process and reducing the number of elements in the repository keeps the computational expense in check. Finally the $p_{best}$ is updated to a new value. The $p_{best}$ is chosen based on whether the new fitness dominates the current $p_{best}$. If neither dominates the other then any one of the two is chosen at random to be the new $p_{best}$. The iteration is now repeated till it reaches the maximum number of iterations set by the user.

Table 3.4: MOPSO suggested range for tuning parameters [17]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of particles</td>
<td>20 - 80</td>
</tr>
<tr>
<td>Number of iterations</td>
<td>80 - 120</td>
</tr>
<tr>
<td>Number of hypercube divisions</td>
<td>30 - 50</td>
</tr>
<tr>
<td>Repository size</td>
<td>Problem dependent</td>
</tr>
</tbody>
</table>

Table 3.4 lists out the range for parameters as suggested by the inventors of the algorithm. As mentioned earlier, the number of particles though improves the success rate of the algorithm requires more function evaluations overall. If the objective functions have a relatively smooth terrain, less particles can be used. The inventors give no guidelines for the size of the repository since it entirely upon the problem at hand and also depends on the resolution of the Pareto front required by the user. For problems where the function evaluations are simple, a limit on the size of the
repository is necessary to speed up the algorithm’s processing time. For function evaluations that are computationally intensive, the non-dominated point filtering process may have a negligible time component and hence less significant.

### 3.3.5 Example Problem

The MOPSO was tested with a generic test function, which is a set of two objective functions with two variables as shown below [158]:

\[
\begin{aligned}
\min_{x \in [-7,4]} \quad & f_1(x, y) = x^2 - y \\
& f_2(x, y) = -0.5x - y - 1 \\
\text{s.t} & \quad g_1(x, y) \leq 6.5 - \frac{x}{6} - y \\
& \quad g_2(x, y) \leq 7.5 - 0.5x - y \\
& \quad g_3(x, y) \leq 30 - 5x - y
\end{aligned}
\]

The settings used for the PSO are tabulated in Table 3.5. The “Absorbing walls” constraint handling technique is adopted in the model as a search space constraint method and the inertia term was ignored when the particle is in the non-feasible region of the objective space [12, 132]. Figure 3.5 shows snapshots of the PSO. It can be seen that the non-dominated points found increases with every iteration and at \( t = 112 \) the repository size is completely filled. Also noticed is that the location of \( g_{best} \) is different for every snapshot highlighting the random assignment at every iteration.
Figure 3.5: Snapshots of the MOPSO at various iterations

(a) $t = 1$, rep. = 4

(b) $t = 10$, rep. = 26

(c) $t = 22$, rep. = 59

(d) $t = 35$, rep. = 74

(e) $t = 75$, rep. = 117

(f) $t = 112$, rep. = 150

Figure 3.5: Snapshots of the MOPSO at various iterations
Table 3.5: PSO parameters used in example problem

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Population Size</td>
<td>50</td>
</tr>
<tr>
<td>$V_{max}$</td>
<td>$0.3 \times 2X_{max}$</td>
</tr>
<tr>
<td>$c_1$ &amp; $c_2$</td>
<td>2 &amp; 2</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.4</td>
</tr>
<tr>
<td>Number of hypercube divisions</td>
<td>35</td>
</tr>
<tr>
<td>Repository size</td>
<td>150</td>
</tr>
</tbody>
</table>
CHAPTER 4
ANALYSIS AND OPTIMIZATION OF THE ORC

In this chapter, a detailed analysis of the model will be performed with respect to the first law and second law of thermodynamics based on points generated from a steady-state Design of Experiments (DOE). The effect of control inputs on ORC parameters such as power output, cycle efficiency will be studied from the DOE.

Later, the MOPSO algorithm previously discussed in Chapter 3 will be made use of to generate a Pareto front for two ORC variables, net power output $P_{\text{net}}$ and system exergy destruction $\dot{E}_d$. The operation of the ORC along the Pareto front will be analyzed with respect to the first and second law of thermodynamics followed by a discussion on selecting a suitable operating point. Then, the Pareto front will be parameterized to allow its use in the single objective optimization (SOO) paradigm. Subsequently, the optimization will be performed for an off-design engine operating point for exhaust gas conditions at 57 mph and differences with respect to the 65 mph condition will be discussed.

4.1 Design of Experiments and Analysis

A full factorial DOE was conducted with respect to the control input positions to gain an understanding of the ORC’s behavior at design point exhaust conditions. As
previously explained in Section 2.9, the design point refers to the exhaust gas condition at 65 mph and an ORC operating point that provides reasonable performance at that exhaust gas condition (see Table 4.1 reproduced from Table 2.7)

Over the course of running the DOE some of the constraints detailed in Table 4.2 (reproduced from Table 2.6) might be violated, since not all combinations of control inputs would yield physically valid results. For example, setting the expander speed in the low end of its valid range when there is low exhaust bypass and high pump speed would cause the evaporator pressure to exceed its upper bounds. While one can setup the boundaries of the DOE with an educated guess of when the model constraints will be violated and avoid running the simulations with those combinations of control inputs, there will still be violations of the constraints and those results can be filtered out of the analysis during post processing. The DOE consists of a total 108 data points of which 83 were within the constraints.¹

Table 4.1: Characterization of design point

<table>
<thead>
<tr>
<th>Vehicle States:</th>
<th>Control Input Positions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Speed</td>
<td>Pump Speed</td>
</tr>
<tr>
<td>Vehicle Speed</td>
<td>65 mph</td>
</tr>
<tr>
<td>Exhaust Temperature</td>
<td>Pump Speed</td>
</tr>
<tr>
<td>Exhaust Temperature</td>
<td>Expander Speed</td>
</tr>
<tr>
<td>Exhaust Temperature</td>
<td>651°C</td>
</tr>
<tr>
<td>Exhaust Mass Flow Rate</td>
<td>Expander Speed</td>
</tr>
<tr>
<td>Exhaust Mass Flow Rate</td>
<td>0.0325 kg/s</td>
</tr>
<tr>
<td>Exhaust Mass Flow Rate</td>
<td>Exhaust gas bypassed</td>
</tr>
<tr>
<td>ORC Target States:</td>
<td>Coolant Pump</td>
</tr>
<tr>
<td>Evaporator Outlet Temperature</td>
<td>Coolant Pump Speed</td>
</tr>
<tr>
<td>Evaporator Outlet Temperature</td>
<td>483°C</td>
</tr>
<tr>
<td>Refrigerant Mass Flow Rate</td>
<td>Coolant Pump Speed</td>
</tr>
<tr>
<td>Refrigerant Mass Flow Rate</td>
<td>0.04 kg/s</td>
</tr>
<tr>
<td>Evaporator Pressure</td>
<td>Coolant Flow Rate</td>
</tr>
<tr>
<td>Evaporator Pressure</td>
<td>21 bar</td>
</tr>
<tr>
<td>Condenser Pressure</td>
<td>Coolant Flow Rate</td>
</tr>
<tr>
<td>Condenser Pressure</td>
<td>2.5 bar</td>
</tr>
<tr>
<td></td>
<td>104.2 g/s</td>
</tr>
</tbody>
</table>

¹During the evolution of the ORC model, the first DOE attempted with an older version of the model had over 900 data points evaluated
Table 4.2: ORC physical constraints

<table>
<thead>
<tr>
<th>State</th>
<th>Limits</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>System:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Refrigerant Mass Flow Rate</td>
<td>$0.02 &lt; \dot{m}_{ref} &lt; 0.07$</td>
<td>kg/s</td>
</tr>
<tr>
<td>Evaporator Pressure</td>
<td>$15 &lt; p_{evap} &lt; 25$</td>
<td>bar</td>
</tr>
<tr>
<td>Evaporator Outlet Temperature</td>
<td>$T_{evapout,sh} &lt; 235$</td>
<td>°C</td>
</tr>
<tr>
<td>Evaporator Inlet Subcool</td>
<td>$\Delta T_{evapin,sc} &lt; -5$</td>
<td>°C</td>
</tr>
<tr>
<td>Evaporator Outlet Superheat</td>
<td>$\Delta T_{evapout,sh} &gt; 5$</td>
<td>°C</td>
</tr>
<tr>
<td>Condenser Outlet Subcool</td>
<td>$\Delta T_{condout,sc} &lt; -5$</td>
<td>°C</td>
</tr>
<tr>
<td><strong>Control Input:</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Feed pump Speed</td>
<td>$175 \leq N_{fp} \leq 675$</td>
<td>rpm</td>
</tr>
<tr>
<td>Expander Speed</td>
<td>$1400 \leq N_{e} \leq 4400$</td>
<td>rpm</td>
</tr>
<tr>
<td>Exhaust Bypass Position</td>
<td>$0 \leq x_{bp} \leq 1$</td>
<td>-</td>
</tr>
</tbody>
</table>

Figure 4.1 shows a scatter plot of expander power output $P_e$ and cycle efficiency $\eta_{ORC}$ normalized with respect to their maximum values. Notice in (Figure 4.1a), an
increase in power output as the exhaust bypass $x_{BP}$ decreases. Naturally, as more exhaust flows through the evaporator, the refrigerant flow rate has to be increased to absorb the surplus heat and also prevent the refrigerant going over its temperature limits, thus the increase in $N_{fp}$ follows the reduction in $x_{BP}$ as observed in Figure 4.1b. With increase in refrigerant flow rate, if the expander speed $N_e$ is constant, there will be a pressure buildup at the evaporator and hence $N_e$ too has to be increased to maintain constant pressure. The relation between the control inputs can be summarized as follows; to maintain constant evaporator outlet temperature $T_4$ and pressure $p_H$, a reduction in exhaust bypass $x_{BP}$ should be followed by an increase in feed pump speed $N_{fp}$ and expander speed $N_e$. Doing so will also result in increased power output $P_e$ from the expander as per the equation below:

$$P_e = m_{ref}(h_4 - h_5)$$ (4.1.1)

The net power output $P_{net}$ is calculated by subtracting the power consumed by the feed pump and coolant pump.

$$P_{net} = P_e - P_{fp} - P_{cp}$$ (4.1.2)

The power consumed by both the pumps is negligible though when compared to the expander power.
Figure 4.2: Normalized $P_e$, $Q_{evap}$ and $\eta_{ORC}$ results from DOE with progression of $N_e$ detailed. Each line connects data points with identical $x_{BP}$ and $N_{fp}$ but different $N_e$.

Delving into the effect of expander speed $N_e$ on the system, each line in Figure 4.2 connects data points which have identical $x_{BP}$ and $N_{fp}$ but different $N_e$. On each line in the plot, moving up (to the right in Figure 4.2c) reduces expander speed $N_e$. In Figure 4.2a, reducing $N_e$ causes an increase in both cycle efficiency $\eta_{ORC}$ and expander power $P_e$. The decrease in $N_e$ restricts the flow of refrigerant leading to an increase in $p_H$ which causes a higher enthalpy drop across the expander, resulting in higher $P_e$ and cycle efficiency $\eta_{ORC}$. The heat input through the evaporator $Q_{evap}$ is insensitive to variation in $N_e$ as observed in Figures 4.2b and 4.2c.
Figure 4.3: Effect of $T_4$ and $p_H$ on a) cycle efficiency [-] (normalized w.r.t maximum efficiency in plot) b) Fraction of total heat supplied to the refrigerant by recuperator [-].

Analyzing the effects of evaporator pressure $p_H$ and evaporator outlet temperature $T_4$ on the ORC, it is observed from Figure 4.3a that the cycle efficiency increases with $T_4$ and $p_H$. While the effect of $p_H$ on the cycle efficiency can be explained by an increase in enthalpy drop across the expander as mentioned previously, the relation between $T_4$ and $\eta_{ORC}$ is a due to both the thermodynamic property of the fluid and also the behavior of the recuperator. The enthalpy drop across the expander increases with $T_4$, resulting in higher $P_e$. But with increase in $T_4$, $T_5$ increases too which also means that there is higher heat transfer at the recuperator resulting in higher temperature of the refrigerant entering the evaporator ($T_3$). The higher inlet temperature at the evaporator reduces the temperature differential between the two streams entering the evaporator and thus heat absorbed by the refrigerant is lesser. This means that the contribution of the evaporator to the total heat absorbed by refrigerant at the high pressure side of the ORC, i.e. between states 2 and 4 is diminished. The higher quantity of heat recycled into the system by the recuperator
results in the higher $\eta_{ORC}$. The fraction of heat supplied by the recuperator $\dot{Q}_{rec,frac}$ is calculated as follows:

$$\dot{Q}_{rec,frac} = \frac{\dot{Q}_{rec}}{\dot{Q}_{rec} + \dot{Q}_{evap}}$$

Figure 4.3b shows $\dot{Q}_{rec,frac}$ at various $T_4$ and $p_H$. Here, $\dot{Q}_{rec,frac}$ clearly increases with $T_4$ while it slightly reduces with increase in $p_H$.

It must be noted that the relation between evaporator superheat temperature and cycle efficiency is opposite to that observed in non-recuperated ORC’s that use R245fa where the cycle efficiency decreases with increase in evaporator outlet temperature.

### 4.1.1 Second Law Analysis

The ORC is now evaluated through the eyes of the second law of thermodynamics by analyzing exergy flows in the system. According to the second law of thermodynamics, when a system is of higher temperature and pressure than its environment, useful work can be extracted from the system in the process of bringing the system in equilibrium with the environment. The useful work that potentially can be extracted during this process is called Exergy. Consider an automobile engine for example, roughly one-third of the fuel energy goes through the exhaust and another third is expended through the cooling system, but the potential amount of work that can be extracted from the exhaust is a lot more than the coolant because the exhaust is at a far higher temperature than the coolant. While a first law analysis of the energy flows would not highlight this, in the second law analysis it becomes evident that the exhaust flow is a more attractive source for recovering useful work since the exergy of the exhaust would be higher than the coolant. Hence, an exergy analysis allows
evaluating multiple energy sources and helps identify the most attractive one. Exergy can be calculated from the equation below:

\[
E = (U - U_o) + p_o(V - V_o) - T_o(S - S_o)
\]

\[
= (H - H_o) - T_o(S - S_o)
\]

(4.1.3)

where the subscript ‘o’ refers to the state of the environment, called as the dead state. The dead state will be assumed to be air at 298 K at 1 bar.

Exergy is not a conserved quantity and can be destroyed. Exergy destruction happens through irreversibilities in the system such as heat transfer over a finite temperature difference, flow mixing, friction, unrestrained expansions of gases and liquids, etc. [18]. If the difference in exergy between two states in any process, say 1 and 2 need to be calculated the equation may be expressed as:

\[
E_1 - E_2 = ((H_1 - H_o) - T_o(S_1 - S_o)) - ((H_2 - H_o) - T_o(S_2 - S_o))
\]

\[
= (H_1 - H_2) - T_o(S_1 - S_2)
\]

(4.1.4)

where the term \(T_o(S_1 - S_2)\) represents the irreversibilities introduced, i.e entropy generated between the states and hence is the exergy destroyed. Even though the exergy is not a conserved quantity, the exergy changes, transfers and destruction have to be accounted for and this can be achieved through an exergy balance equation. In the plant model of the ORC being studied, it is more convenient to use the control volume form of the exergy balance equation expressed below [18],

\[
\frac{dE_{cv}}{dt} = \sum_j \left(1 - \frac{T_o}{T_j}\right) \dot{Q}_j - \left(\dot{W}_{cv} - p_o \frac{dV_{cv}}{dt}\right) + \sum_i \dot{m}_i e_i - \sum_e \dot{m}_e e_e - \dot{E}_d
\]

(4.1.5)

where the subscript ‘j’ represents heat transfer surfaces and ‘i’ and ‘e’ represent the inlets and outlets for mass flow. \(e\) is the specific exergy. \(\dot{E}_d\) is the exergy destruction rate within the control volume. \(\dot{E}_d\) can either be zero or positive but never
negative. The first term on the right represents exergy lost through heat transfer to the surroundings and the second term represents work done on or by the control volume.

At steady state, equation 4.1.5 can be simplified to,

\[ 0 = \sum_j \left( 1 - \frac{T_o}{T_j} \right) \dot{Q}_j - \dot{W}_{cv} + \sum_i \dot{m}_i e_i - \sum_e \dot{m}_e e_e - \dot{E}_d \quad (4.1.6) \]

The above equation will be used to calculate the exergy destruction in each component of the ORC plant model. Since heat transfer to the surroundings is ignored in the ORC plant model, the first term on the right hand side of the above equation will always be zero.

A term that can be used to measure the second law conversion efficiency of a component is the \textit{exergy efficiency} (or \textit{rational efficiency}). The exergy efficiency provides a way to understand how much of the exergy flowing into the system is being effectively used and is analogous to the first law efficiency. The exergy efficiency can be calculated by dividing all the exergy flowing out of a control volume by the exergy flowing into the control volume as shown below [18],

\[ \text{exergy efficiency } \Psi = \frac{\sum \dot{E}_{out}}{\sum \dot{E}_{in}} = 1 - \frac{\dot{E}_d}{\sum \dot{E}_{in}} \quad (4.1.7) \]

\subsection*{4.1.2 Exergy Destruction in the ORC System}

In the ORC plant model, seven points have been identified as sources of exergy destruction and are listed in Table 4.3 along with the equations required to calculate the exergy destruction at these points. Irreversibilities due to pipe friction and heat transfer to the surroundings from all components have been ignored in the plant model and hence do not appear in the exergy analysis. The total exergy destruction in the system is the sum of the exergy destruction from all the components,

\[ \dot{E}_d = \dot{E}_{d,\text{evap}} + \dot{E}_{d,\text{mix}} + \dot{E}_{d,e} + \dot{E}_{d,\text{rec}} + \dot{E}_{d,\text{cond}} + \dot{E}_{d,fp} + \dot{E}_{d,cp} \quad (4.1.8) \]

115
Table 4.3: Exergy destruction equations for individual components of the ORC [18]

<table>
<thead>
<tr>
<th>Component</th>
<th>Exergy destruction equations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaporator</td>
<td>$\dot{E}<em>{d,\text{evap}} = \dot{m}<em>E E_2 (e</em>{E2} - e</em>{E3}) - \dot{m}_\text{ref} (e_4 - e_3)$</td>
</tr>
<tr>
<td>Recuperator</td>
<td>$\dot{E}<em>{d,\text{rec}} = \dot{m}</em>\text{ref} (e_6 - e_5) - \dot{m}_\text{ref} (e_3 - e_2)$</td>
</tr>
<tr>
<td>Condenser</td>
<td>$\dot{E}<em>{d,\text{cond}} = \dot{m}</em>\text{ref} (e_5 - e_1) - \dot{m}<em>C (e</em>{C3} - e_{C2})$</td>
</tr>
<tr>
<td>Exhaust downstream mixing</td>
<td>$\dot{E}_{d,\text{mix}} = \dot{m}<em>E E_3 e</em>{E3} - (\dot{m}<em>E E_2 e</em>{E2} + \dot{m}<em>B e</em>{Eb})$</td>
</tr>
<tr>
<td>Expander</td>
<td>$\dot{E}<em>{d,e} = \dot{m}</em>\text{ref} T_o (s_5 - s_4)$</td>
</tr>
<tr>
<td>Feed pump</td>
<td>$\dot{E}<em>{d,fp} = \dot{m}</em>\text{ref} T_o (s_2 - s_1)$</td>
</tr>
<tr>
<td>Coolant pump</td>
<td>$\dot{E}<em>{d,cp} = \dot{m}<em>C T_o (s</em>{C2} - s</em>{C1})$</td>
</tr>
</tbody>
</table>

The exergy of the exhaust gas at design point conditions, ~1 bar, 924 K and 0.0325 kg/s is 9.633 kW. With that being said, the exergy destruction in each component for the ORC at the design point is tabulated in Table 4.4. From Table 4.4, it is evident that the evaporator contributes the most towards exergy destruction in the system.

Irreversibilities in heat exchangers may be attributed to three reasons, namely, heat transfer between fluids over a finite temperature difference, pressure losses due to fluid friction and heat dissipation to the surroundings [159]. Exergy is destroyed in the first two modes and exergy is lost to the surroundings in the third. In the ORC model, dissipation to the surroundings in heat exchangers is ignored due to its negligible effect. The majority of the exergy destruction occurs during heat transfer to the fluid of lower temperature. In the ORC model, since the temperature difference between the two fluid streams in the recuperator and condenser is not as high as the
Table 4.4: Exergy data for the ORC at design point

<table>
<thead>
<tr>
<th>Component</th>
<th>Exergy Destruction</th>
<th>Exergy Destruction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Evaporator</td>
<td>2.490 [kW]</td>
<td>53.0 %</td>
</tr>
<tr>
<td>Exhaust downstream mixing</td>
<td>0.900 [kW]</td>
<td>19.3 %</td>
</tr>
<tr>
<td>Condenser</td>
<td>0.430 [kW]</td>
<td>9.2 %</td>
</tr>
<tr>
<td>Expander</td>
<td>0.420 [kW]</td>
<td>9.0 %</td>
</tr>
<tr>
<td>Recuperator</td>
<td>0.415 [kW]</td>
<td>8.8 %</td>
</tr>
<tr>
<td>Feed pump</td>
<td>0.018 [kW]</td>
<td>~0.0 %</td>
</tr>
<tr>
<td>Coolant pump</td>
<td>~0.000 [kW]</td>
<td>~0.0 %</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>4.670 [kW]</strong></td>
<td><strong>100 %</strong></td>
</tr>
</tbody>
</table>

evaporator and also due to the higher average temperature of fluids involved in the evaporator, there is relatively higher exergy destruction in the evaporator.

The mixing point of the bypassed exhaust stream and the outlet of the evaporator is also a source of high exergy destruction. Irreversibilities in the mixing process occurs due to energy transfer between the two converging streams when they are not in thermal equilibrium [14]. Modes of irreversibilities in the mixing process such as heat transfer and viscous dissipation are ignored and intermingling of molecules of different species does not apply in this case [14, pg.138].

Irreversibilities during the expansion process in the expander can be attributed to internal leakage, friction, heat transfer and throttling losses at the outlet due to under and over expansion [160,161].
An exergy analysis on the individual components of the ORC model is performed using the data points obtained from the DOE. Figure 4.4 shows the exergy destruction
for the each component of the ORC plotted with respect to the three control inputs and Figure 4.5 displays the same information normalized with respect to the total exergy destruction $\dot{E}_d$ in the system. In other words, the percentage contribution of each component.

The plots highlight the significance of the evaporator’s contribution to the total exergy destruction when compared to the other components. The evaporator’s percentage contribution to the overall exergy destruction increases with $N_{fp}$ and decrease in $x_{BP}$. The exhaust mixing shows a trade off relation with $N_{fp}$ and $N_e$ that will be discusses later. $\dot{E}_d$ seems to correlate less with $N_e$ and hence has a lesser influence on the exergy destruction when compared to the other control inputs. The exergy destruction in the evaporator is clearly dependent on the refrigerant flow rate since it correlates well with the feed pump speed $N_{fp}$. An analysis on the exergy destruction at a component level for the evaporator, exhaust mixing point and the expander will be discussed below.

**Evaporator**

The treatment of exergy by the evaporator can be analyzing through its exergy efficiency, which can be expressed as [14],

$$
\Psi_{evap} = \frac{\dot{m}_{exh} (e_4 - e_3)}{\dot{m}_{ref} (e_{E2} - e_{E3})}
$$

(4.1.9)
As mentioned previously, the exergy destruction in the evaporator is mainly through heat transfer over a finite temperature difference. The exergy efficiency of the evaporator is plotted with respect to the difference in its inlet temperatures in Figure 4.6. It shows that $\Psi$ decreases as the inlet temperature difference between the exhaust and refrigerant increase. This is expected, since there is higher exergy destruction when heat is transferred to a lower temperatures. In the same figure, $\Psi$ is also shown to improve with increase in pressure. This may be attributed to two reasons, the first being that higher pressure results in a higher saturation temperature, raising the average temperature of the refrigerant in the evaporator. The second reason being, the increase in slope of the constant pressure line as pressure increases in the h-s plot for R245fa which means that there is a higher enthalpy gain per unit gain of entropy at high pressures during constant pressure heat addition in the heat exchanger thus reducing irreversibilities.
Exhaust Mixing

With regards to exergy destruction during exhaust mixing, $\dot{E}_{d,\text{mix}}$ peaking at $\sim 50\%$ exhaust bypass in Figure 4.5 can be explained through the relationship between exergy and enthalpy at constant pressure as seen in Figure 4.7. Assuming negligible pressure drop across the exhaust circuit, the exhaust flow can be represented by the iso-pressure line for 1 bar in Figure 4.7. The non-linear nature of this line is what causes the variation in exergy destruction at different $x_{BP}$ values. The enthalpy of the output stream $E4$ can be calculated by taking a weighted average of the two incoming streams $E3$ and and $Eb$:

$$h_{E4} = \frac{\dot{m}_{E3}h_{E3} + \dot{m}_{Eb}h_{Eb}}{\dot{m}_{E3} + \dot{m}_{Eb}}$$  \hspace{1cm} (4.1.10)$$

where $h$ is the specific enthalpy of air.

The behavior can be illustrated with Figure 4.8. In Figure 4.8, let the curved line represent the relationship between exergy and enthalpy for air at 1 bar. As the exhaust bypass increases, the enthalpy of the mixed exhaust $E4$ moves from $E3$ to
Figure 4.8: Illustration of exergy loss through exhaust mixing

$E_b$ along the curved line. The exergy destruction is the length of the line $E_4' - E_4$. Moving away from $E_3$ the length of the line $E_4' - E_4$ increases up to a point roughly midway between $E_3$ and $E_b$, after which it starts decreasing, which is the same as observed in the Figure 4.4 for exhaust mixing.

**Expander**

The exergy efficiency of the expander can be defined as [14],

$$\Psi_e = \frac{P_e}{\dot{E}_4 - \dot{E}_5} \quad (4.1.11)$$
The exergy efficiency is plotted in Figure 4.9a. The plot shows that the exergy efficiency increases with increase in expander inlet temperature $T_4$. Ideally, the expander would isentropically expand the fluid but due to practical reasons this is not the case. The irreversibilities in the expansion process will cause the expander to operate with an isentropic efficiency less than 100%, generating entropy during the expansion process resulting in the fluid exiting the expander at a temperature higher than the ideal case. The higher temperature of the fluid at the exit in the non-ideal case when compared to isentropically expanded fluid means that the potential work lost can be salvaged in a downstream process because the higher temperature makes it more attractive. The difference in enthalpy between the ideal and non-ideal case is called frictional reheat [14]. This frictional reheat is the reason why exergy efficiency is higher than the isentropic efficiency. This can be visualized in the T-S diagram of the expansion process shown in Figure 4.9b. The fluid undergoes non-ideal expansion between 4 and 5 with 4-5' representing isentropic expansion. The exergy lost is
signified by the cross hatched rectangle $abcd$. $5'5ba$ represents the frictional reheat which can be though of as isobaric heat addition. As $T_4$ increases, the points follow the isobar which causes the size of the frictional reheat area $5'5ba$ to increase thus improving the exergy efficiency. Increasing $p_H$ though would cause the point 4 to move laterally to the left thus reducing the size of $5'5ba$ resulting in lower exergy efficiency. The expander in the ORC plant model follows these trends as seen in Figure 4.9a. [14, ch.4] provides a more detailed analysis of the exergy analysis of the expander.

4.1.3 The Optimization Problem

From the analysis conducted so far, especially surrounding Figure 4.1 and Figure 4.4, it has been established that both $P_e$ (henceforth substituted with $P_{net}$) and $\dot{E}_d$ rise with $\dot{m}_{ref}$.

![Figure 4.10: System exergy destruction vs net expander power](image)

Figure 4.10 shows total exergy destruction in the system $\dot{E}_d$ plotted against the
inverse of net power output of the ORC. The plot shows that a trade-off exists between the two quantities, i.e. both quantities cannot be optimized at the same time and optimizing one quantity must result in the deterioration of the other, in other words, a Pareto front exists. The MOPSO algorithm described in Chapter 3 will be employed to generate the Pareto front. The Pareto front will provide a complete picture of the trade-off relationship between the two quantities which allows the system designer to make a well informed decision.

While the importance of maximizing $P_{\text{net}}$ is clear, it may not be clear as to the purpose of utilizing $\dot{E}_d$ as a parameter to be optimized. The idea behind the use of $\dot{E}_d$ in the context of this thesis is to balance against system net power $P_{\text{net}}$ to prevent the ORC from violating the control input constraints on the system. The lower the exergy destruction, the further away from the control input constraints, will the system operate. Exergy destruction $\dot{E}_d$ is a more generalized quantity that may be applied to other thermal systems.

4.2 Multi-Objective Optimization with PSO

In this section the MOPSO algorithm will be utilized to generate the Pareto front between net power output and exergy destruction. The MOO problem may be defined as,

$$\min_{x \in \Omega} \left( \dot{E}_d \frac{1}{P_{\text{net}}} \right)$$

$$g_i(x) \leq 0, \quad i = 1, 2, ..., 8$$

where $\Omega$ represents control input range and $g_i(x)$ the system constraints as defined in Table 4.2. The exhaust gas conditions are set constant to the design point values
corresponding to 65 mph. The process of setting up the MOPSO will be detailed below.

4.3 Setup

The MOPSO algorithm was coded in MATLAB® with the GT-SUITE ORC plant model called from within a MATLAB® function. The function also extracts the objectives and constrained parameters from the GT-SUITE model and passes it to the PSO algorithm to be evaluated.

The input variables for the PSO are $N_{fp}$, $N_{e}$ and $x_{BP}$ and objective functions are $P_{net}$ and $\dot{E}_{d}$. The objective $P_{net}$ needs to be maximized while $\dot{E}_{d}$ has to be minimized.\(^2\)

Constraints need to be imposed on both the search space and the model parameters to ensure that the results obtained are feasible and physically valid. The search space constraints are taken from the Control Input section of Table 2.6. The “Absorbing Walls” method (see pg. 90) is adopted to restrict the motion of the particles within the search space. The positions of the particles are moved to the boundaries if they stray outside the search space. The adjustment occurs immediately after the new positions are calculated as shown in Algorithm 2. Constraints on the system parameters are imposed after evaluating the objective function, in this case the GT-SUITE ORC plant model. If the parameters for any particle are not within the constraints set in the system section of Table 2.6 then the particle is flagged to be excluded from consideration for pbest or gbest assignment. Also, the inertia term in the velocity equation is disabled for the next iteration, until the particle is in the feasible region again [132].

Table 4.5 lists out the parameters set in the MOPSO. The acceleration coefficients

\(^2\)Since the PSO script was written to maximize a variable, $-\dot{E}_{d}$ flips the goal of exergy destruction into maximization but $\frac{1}{\dot{E}_{d}}$ could also serve the same purpose.
$c_1 \& c_2$ were adopted from the values recommended in [127]. The inertia weight $\omega$ was kept to what was recommended in the original MOPSO algorithm [17]. The population size was restricted to only 6 particles due to the availability of limited software licenses and with each simulation taking about $\sim$ 5-20 minutes, deciding that lesser particles in the swarm would be more efficient because the iteration would progress faster providing better guidance to the swarm. The size of the repository was not restricted because the function evaluations were far more computationally intensive when compared to the non-dominated point filtering process.

Algorithm 2 shows the MOPSO algorithm updated with the constraint handling mechanisms mentioned earlier. A feasibility check is done after every iteration and all infeasible particles are flagged to prevent their consideration for $p_{best}$ and $g_{best}$. The initialization process for the particles will be repeated until at least 1 feasible particle is obtained to make sure that there is at least one feasible particle that can be assigned $g_{best}$ initially. Alternatively, multiple loops of initialization process in search of a feasible starting point can be avoided by specifying a set of start points where at least one position is known to be feasible. Since the optimization process is computationally intensive, a mechanism to preload a set of repository entries was added to allow restarting the optimization process in case the optimizer has to be stopped midway for any reason.
Algorithm 2 MOPSO for ORC

1: repeat
2: Initialize swarm position POS
3: Initialize VEL = 0
4: Evaluate all particles
5: Feasibility check
6: until Atleast 1 particle is feasible
7: Find non-dominated solutions and add their positions into repository REP
8: Classify particles in REP into hypercubes based on their position in the objective function space.
9: Initialize PBEST = POS
10: while ITER < MAXITER do
11: Select REP[h] for the current iteration.
12: \[ VEL = W \times VEL + R_1 \times (PBEST - POS) + R_1 \times (REP[h] - POS) \]
13: POS = POS + VEL
14: Restrict particle position within the searchspace.
15: Evaluate all particles
16: Feasibility check
17: Update repository and hypercube. If repository is full then remove particles at random starting with the least populated hypercube.
18: \[ \triangleright \] Continued in the next page
Algorithm 2 MOPSO for ORC (continued)

19: if POS dominates PBEST then  
20: \hspace{1cm} \text{PBEST} = \text{POS}  
21: \hspace{1cm} \text{else if} \text{ PBEST dominates POS then}  
22: \hspace{1cm} \text{no change in PBEST}  
23: \hspace{1cm} \text{else} \text{ neither dominates}  
24: \hspace{1cm} \text{pick either one at random as the new PBEST}  
25: \hspace{1cm} \text{end if}  
26: \hspace{1cm} \text{ITER} = \text{ITER} + 1  
27: \text{end while}

Table 4.5: Summary of PSO parameters for the Pareto optimization

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$c_1$ &amp; $c_2$</td>
<td>2.8 &amp; 1.3</td>
</tr>
<tr>
<td>$\omega$</td>
<td>0.4</td>
</tr>
<tr>
<td>Population Size</td>
<td>6</td>
</tr>
<tr>
<td>$V_{max}$</td>
<td>$1 \times 2X_{max}$</td>
</tr>
<tr>
<td>Number of hypercube divisions</td>
<td>20</td>
</tr>
<tr>
<td>Repository size</td>
<td>$\infty$</td>
</tr>
</tbody>
</table>
4.4 Results

The optimizer was executed on the GT-SUITE model with exhaust gas conditions set to the design point (see Table 2.7). The MOPSO was allowed to run for about 100 iterations yielding 22 non-dominated (ND) points. In this section, the Pareto front will first be parameterized followed by the analysis of the points on the Pareto front and suggestion of an operating point.

4.4.1 Parameterization of the Pareto Front

Figure 4.11: Pareto optimal points generated from the MOPSO overlaid on DOE data points

Figure 4.11 displays the ND points obtained from the optimization process with Table 4.10 displaying relevant system parameters of the ND points. The ND points in the figure are linked together to form the Pareto front which is overlaid on the data points from the DOE. All the points were normalized in each dimension with respect to the maximum value on the Pareto front in that dimension. Observe that the Pareto
front formed by the ND points obtained is not smooth which can be attributed to the size of the swarm being very small and hence being less effective than expected but the ND points still enclose all the DOE data points which serves as a sanity check for the MOPSO algorithm and proves that the algorithm is effective enough. Also, in Figure 4.11, observe the design point — highlighted in green — seems to be located on the Pareto front, near the central region. The lowest $\dot{E}_d$ is at $\approx 40\%$ of the maximum on the Pareto Front and lowest $P_{net}$ extends to $\approx 25\%$ of it’s maximum.

As a first step towards parameterization, the Pareto front was curve fit with a rational function shown below,

$$f(x) = \frac{p_1 x^2 + p_2 x + p_3}{x + q_1}$$

(4.4.1)

where

$$p_1 = 0.5103 \quad p_2 = -0.7839 \quad p_3 = 0.8302 \quad q_1 = 0.3854$$

$$x = \frac{P_{ref}}{P_{net}} \quad f(x) = \frac{\dot{E}_d}{\dot{E}_{d,ref}}$$

$\dot{E}_{d,ref}$ and $P_{ref}$ are the normalization factors applied to the objective functions which are taken from point 1 and 22 respectively in Table 4.10.

With this curve fit function $f(x)$, a weighted aggregate objective function is constructed as shown below:

$$F(x) = \alpha x + (1 - \alpha) f(x)$$

(4.4.2)

with $\alpha \in [0, 1]$. Suppose the two ORC plant model outputs were to be expressed as
functions of the form $\dot{E}_d(N_{fp}, N_e, x_{BP})$ and $P_{net}(N_{fp}, N_e, x_{BP})$, an aggregate objection function can be created,

$$G(x) = \alpha \frac{\dot{E}_d(N_{fp}, N_e, x_{BP})}{\dot{E}_{d,ref}} + (1 - \alpha) \frac{P_{ref}}{P_{net}(N_{fp}, N_e, x_{BP})} \tag{4.4.3}$$

Since the Pareto front $f(x)$ encompasses the set of all optimized solutions for $G(x)$ for all possible weights $\alpha$, the optimized result must be located on the Pareto front. Therefore, solving $F(x)$ with the same weight $\alpha$ will result in arriving at the same point on the Pareto front as solving $G(x)$ with $\alpha$. Though it should be pointed out that since $f(x)$ is a curve fit on the Pareto front, the actual result will not be exactly at the same point but roughly near the optimized location.

![Figure 4.12: $F(x)$ with $\alpha$ sweep](image)

Figure 4.12 shows multiple lines of $F(x)$ with each line representing a different value for weight $\alpha$. The highlighted and labeled points indicate the location of the minimum for the line on which the point is situated and the label represents the weight $\alpha$ for the corresponding line. The minimum is at the ends for a few lines on Figure 4.12 because the constraint on the limits of $x$ exclude the actual minimum of
\[ F(x) \] for that particular \( \alpha \) from being reached. Now that the point \( x \) at which the minimum occurs for each weight \( \alpha \) is known, this information is employed to overlay the \( \alpha \) at the coordinates \((x, f(x))\) on the Pareto front as shown in Figure 4.13.

![Figure 4.13: Pareto front at 65 mph with \( \alpha \) overlaid](image)

Supposing a point \((x_0, f(x_0))\) on the Pareto front is already chosen as the desired operating point. To find the weight \( \alpha \) corresponding to that point, a bilevel optimization problem has to be solved,

\[
\alpha^* = \arg \min_{\alpha \in [0,1]} Y(x, \alpha)
\]

\[
Y(x, \alpha) = x_0 - x^*
\]

\[
x^* \in \arg \min_{x \in [x_{\min}, 1]} F(x)
\]

\(x_{\min}\) is the minimum value of \( x \) on the Pareto front, i.e. the value of \( x \) that corresponds to \( f(x) = 1\).
4.4.2 Pareto Analysis

Figure 4.14 displays the three dimensional search space that consists of the control inputs $N_p$, $N_e$ and $x_{BP}$. The red wireframe box enveloping the data points signifies the search space constraints (see Table 2.6). The design point (green dot) is located roughly in the middle of the cuboid. Point 1 is on the edge of both the constraints of $x_{BP}$ and $N_e$; and also close to the constraint for $N_{fp}$. Moving away from point 1, the control inputs roughly follow the path of the cuboid’s diagonal, moving away from the boundary and finally intersecting the face of the cuboid at point 17 on the other side that pertains to the lower limit of the expander speed $N_e$. Though $N_e$ is prevented from going lower than 1400 rpm, $N_{fp}$ keeps reducing and $x_{BP}$ keep increasing. Hence, if $\dot{m}_{ref}$ drops, $p_H$ will also start to reduce. At the end of the Pareto front (point 22), the system reaches the limit of $\dot{m}_{ref}$.
Figure 4.15 shows the three most pertinent system constraints, $T_4$, $\dot{m}_{ref}$ and $p_H$ displayed in a form similar to Figure 4.14. Here too, the design point is in the center of the cuboid. Point 1 is located near the edge of the cuboid that indicates the constraints of $p_H$ and $\dot{m}_{ref}$. Advancing from point 1, the ND points move away from the borders of the cuboid.
The parameters $T_4$, $\dot{m}_{ref}$ and $p_H$ are displayed separately in Figure 4.16a along with the two objective functions $P_{net}$ and $\dot{E}_d$. In the figure, moving from left to right, the refrigerant flow rate $\dot{m}_{ref}$ decreases steadily while $p_H$ maintains its position on the limit. The ND points stay close to the $p_H$ boundary until about point 17 where the points move away from the limit. This is because the expander speed reaches its lower limit which means that the maximum pressure cannot be reached beyond this point. The evaporator outlet temperature $T_4$ seems to be oscillating between 470 K and 480 K.

Figure 4.16b also highlights the fact that all three control inputs have also reached the limits of their operation at the point corresponding to maximum power (point 1). This poses a problem since operating the control inputs at their limits is not preferable.
4.4.3 Selection of Operating Point

While extracting maximum power $P_{net}$ from the system is always preferred, operation of the ORC at a safe distance away from the control input constraints must also be considered. Point 8 from the 65 mph case would be a suggested location since this point provides 83% of the peak power while at the same time being reasonably away from the constraints as shown in Figure 4.16b. The selected point also produces 50% more power than the design point.

The corresponding weight $\alpha$ for the point 8 could be found using the method outlined in section 4.4.1 which would be $\sim 0.65$. This weight may be input into Equation 4.4.3 which can be used in an optimal control scheme.
Table 4.6: Summary of parameters for Pareto optimal points at 65 mph

<table>
<thead>
<tr>
<th>Point</th>
<th>$P_{net}$</th>
<th>$\dot{E}_d$</th>
<th>$x_{BP}$</th>
<th>$N_{fp}$</th>
<th>$N_e$</th>
<th>$p_H$</th>
<th>$T_4$</th>
<th>$\dot{m}_{ref}$</th>
<th>$Q_{evap}$</th>
<th>$\eta_{ORC}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.00</td>
<td>1.00</td>
<td>0.0</td>
<td>610</td>
<td>4346</td>
<td>24.4</td>
<td>502</td>
<td>66.7</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>0.91</td>
<td>0.99</td>
<td>6.9</td>
<td>600</td>
<td>4213</td>
<td>23.3</td>
<td>481</td>
<td>65.6</td>
<td>0.95</td>
<td>0.95</td>
</tr>
<tr>
<td>3</td>
<td>0.90</td>
<td>0.99</td>
<td>7.3</td>
<td>592</td>
<td>4285</td>
<td>23.0</td>
<td>485</td>
<td>64.8</td>
<td>0.94</td>
<td>0.95</td>
</tr>
<tr>
<td>4</td>
<td>0.89</td>
<td>0.97</td>
<td>10.0</td>
<td>575</td>
<td>4000</td>
<td>23.9</td>
<td>487</td>
<td>62.9</td>
<td>0.91</td>
<td>0.97</td>
</tr>
<tr>
<td>5</td>
<td>0.84</td>
<td>0.97</td>
<td>13.5</td>
<td>570</td>
<td>4012</td>
<td>22.8</td>
<td>475</td>
<td>62.4</td>
<td>0.89</td>
<td>0.94</td>
</tr>
<tr>
<td>6</td>
<td>0.83</td>
<td>0.96</td>
<td>14.1</td>
<td>558</td>
<td>4196</td>
<td>22.1</td>
<td>482</td>
<td>61.1</td>
<td>0.88</td>
<td>0.94</td>
</tr>
<tr>
<td>7</td>
<td>0.83</td>
<td>0.90</td>
<td>19.2</td>
<td>504</td>
<td>3681</td>
<td>24.1</td>
<td>504</td>
<td>55.2</td>
<td>0.82</td>
<td>1.00</td>
</tr>
<tr>
<td>8</td>
<td>0.82</td>
<td>0.89</td>
<td>20.0</td>
<td>500</td>
<td>3500</td>
<td>24.7</td>
<td>502</td>
<td>54.6</td>
<td>0.81</td>
<td>1.00</td>
</tr>
<tr>
<td>9</td>
<td>0.79</td>
<td>0.86</td>
<td>23.9</td>
<td>482</td>
<td>3384</td>
<td>24.6</td>
<td>501</td>
<td>52.6</td>
<td>0.77</td>
<td>1.01</td>
</tr>
<tr>
<td>10</td>
<td>0.70</td>
<td>0.82</td>
<td>31.8</td>
<td>458</td>
<td>2955</td>
<td>24.7</td>
<td>480</td>
<td>49.9</td>
<td>0.71</td>
<td>0.97</td>
</tr>
<tr>
<td>11</td>
<td>0.68</td>
<td>0.80</td>
<td>32.5</td>
<td>432</td>
<td>3500</td>
<td>21.9</td>
<td>500</td>
<td>47.4</td>
<td>0.69</td>
<td>0.97</td>
</tr>
<tr>
<td>12</td>
<td>0.64</td>
<td>0.77</td>
<td>37.6</td>
<td>426</td>
<td>2746</td>
<td>24.4</td>
<td>475</td>
<td>46.4</td>
<td>0.66</td>
<td>0.97</td>
</tr>
<tr>
<td>13</td>
<td>0.60</td>
<td>0.71</td>
<td>42.7</td>
<td>375</td>
<td>2500</td>
<td>24.9</td>
<td>494</td>
<td>40.9</td>
<td>0.60</td>
<td>1.00</td>
</tr>
<tr>
<td>14</td>
<td>0.54</td>
<td>0.70</td>
<td>46.7</td>
<td>384</td>
<td>2300</td>
<td>24.1</td>
<td>459</td>
<td>41.6</td>
<td>0.57</td>
<td>0.94</td>
</tr>
<tr>
<td>15</td>
<td>0.54</td>
<td>0.65</td>
<td>49.0</td>
<td>350</td>
<td>2188</td>
<td>24.6</td>
<td>480</td>
<td>38.1</td>
<td>0.54</td>
<td>0.98</td>
</tr>
<tr>
<td>16</td>
<td>0.47</td>
<td>0.58</td>
<td>55.0</td>
<td>300</td>
<td>2000</td>
<td>23.3</td>
<td>494</td>
<td>32.7</td>
<td>0.47</td>
<td>1.00</td>
</tr>
<tr>
<td>17</td>
<td>0.44</td>
<td>0.56</td>
<td>57.8</td>
<td>297</td>
<td>1575</td>
<td>23.8</td>
<td>473</td>
<td>32.3</td>
<td>0.45</td>
<td>0.97</td>
</tr>
<tr>
<td>18</td>
<td>0.42</td>
<td>0.53</td>
<td>60.0</td>
<td>275</td>
<td>1500</td>
<td>23.3</td>
<td>483</td>
<td>29.9</td>
<td>0.42</td>
<td>0.98</td>
</tr>
<tr>
<td>19</td>
<td>0.35</td>
<td>0.49</td>
<td>65.0</td>
<td>250</td>
<td>1500</td>
<td>20.8</td>
<td>467</td>
<td>27.2</td>
<td>0.38</td>
<td>0.92</td>
</tr>
<tr>
<td>20</td>
<td>0.31</td>
<td>0.46</td>
<td>67.8</td>
<td>237</td>
<td>1400</td>
<td>19.8</td>
<td>457</td>
<td>25.8</td>
<td>0.35</td>
<td>0.89</td>
</tr>
<tr>
<td>21</td>
<td>0.27</td>
<td>0.41</td>
<td>71.5</td>
<td>204</td>
<td>1400</td>
<td>18.0</td>
<td>470</td>
<td>22.1</td>
<td>0.30</td>
<td>0.88</td>
</tr>
<tr>
<td>22</td>
<td>0.25</td>
<td>0.39</td>
<td>73.0</td>
<td>194</td>
<td>1400</td>
<td>17.2</td>
<td>471</td>
<td>21.0</td>
<td>0.29</td>
<td>0.87</td>
</tr>
</tbody>
</table>
Figure 4.17: Cycle analysis of ORC for various points on the Pareto front (Continued in the next page)
Three points on the Pareto front (point 8, 16 and 21) are isolated to compare behavior at various sections of the ORC. The refrigerant flow rate $\dot{m}_{\text{ref}}$ and evaporator pressure $p_H$ reduce as we progress from point 8 to 21.

Now comparing the temperatures at each state of the ORC, the temperature of all states generally rise and lower in tandem except states 1 and 2 which are constant for all points.
4.4.5 Evaporator analysis

(a) Point 8

Figure 4.18: Evaporator profile for various points on the Pareto front (Continued in the next page)

(b) Point 16
Figure 4.18 shows the T-Q diagram of the evaporator for points 8, 16 and 21 on the Pareto front. In the figure, the T-Q diagrams are split into three parts based on the phase of the refrigerant and the following analysis will compare exergy and heat flows in each phase. Table 4.7 displays relevant parameter information pertaining to each phase of the evaporator. The liquid phase has the lowest contribution to heat absorbed by the refrigerant due to the low subcooling of the refrigerant entering the evaporator and also the low temperature difference between the two fluid streams as evidenced by the low Logarithmic Mean Temperature Difference (LMTD) when compared to the other phases. Since the heat of vaporization of the refrigerant varies with pressure, the size of the two phase region is dependent on pressure $p_H$. Point 21 having the lowest pressure has the largest two-phase region of the three points. At points 8 and 16, the gas phase has a highest share of heat transfer while the two phase region has a higher share of heat absorption at point 21. The combination of
higher heat of vaporization and reduced superheating when compared to the other two points may be attributed to the share of heat absorption by the gas phase being lower in point 21.

Heat transfer over a finite temperature difference causes exergy destruction since the receiving medium being of low temperature will have less potential use of the heat absorbed when compared to the transferring medium. Then it is logical to assume that if the temperature difference is higher, exergy destroyed will also be high. The LMTD can be used to quantify the finite temperature difference. The LMTD for each phase is displayed in Table 4.18. When comparing each phase, the contribution of the superheated region to exergy destruction is the highest, even at point 21 where the total share of the heat transfer is lower than the two phase region. Both the combination of heat transfer from a relatively higher temperature and the larger temperature difference, evidenced by the LMTD, contribute to the higher exergy destruction in the gas phase. The exergy efficiency $\Psi$ is also lowest for the gas phase due to the higher temperature difference between the two streams.
Table 4.7: Evaporator parameters for each phase of the refrigerant

<table>
<thead>
<tr>
<th>Property</th>
<th>Unit</th>
<th>Point</th>
<th>Liq.</th>
<th>Two Ph.</th>
<th>Gas</th>
<th>Overall</th>
</tr>
</thead>
<tbody>
<tr>
<td>$Q_{evap}$</td>
<td>[-]</td>
<td>8</td>
<td>0.123</td>
<td>0.352</td>
<td>0.524</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16</td>
<td>0.115</td>
<td>0.385</td>
<td>0.498</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td>0.083</td>
<td>0.477</td>
<td>0.438</td>
<td>1</td>
</tr>
<tr>
<td>$\dot{E}_d$</td>
<td>[-]</td>
<td>8</td>
<td>0.059</td>
<td>0.310</td>
<td>0.630</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16</td>
<td>0.044</td>
<td>0.334</td>
<td>0.620</td>
<td>1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td>0.040</td>
<td>0.414</td>
<td>0.555</td>
<td>1</td>
</tr>
<tr>
<td>LMTD$^a$</td>
<td>K</td>
<td>8</td>
<td>43</td>
<td>163</td>
<td>331</td>
<td>137</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16</td>
<td>22</td>
<td>158</td>
<td>339</td>
<td>100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td>11</td>
<td>169</td>
<td>371</td>
<td>78</td>
</tr>
<tr>
<td>$\Psi$</td>
<td>%</td>
<td>8</td>
<td>88.67</td>
<td>79.40</td>
<td>71.91</td>
<td>76.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16</td>
<td>90.91</td>
<td>79.70</td>
<td>70.92</td>
<td>76.6</td>
</tr>
<tr>
<td></td>
<td></td>
<td>21</td>
<td>90.92</td>
<td>77.85</td>
<td>67.67</td>
<td>74.4</td>
</tr>
</tbody>
</table>

$^a$ For the two phase region, the difference between the average temperature of the exhaust gas and the refrigerant saturation temperature was used instead of the LMTD.
4.5 Pareto Front for Operation of the ORC at Off-Design conditions

Since the exhaust gas will not be constant throughout the vehicle operation, the shape of the Pareto front at off design conditions will also have to be examined. In this section, the Pareto front for the 57 mph case will be generated and compared with the 65 mph case. The MOPSO was run on the ORC plant model with exhaust conditions at 57 mph with settings identical to the 65 mph case. At 57 mph, the exhaust heat rate is relatively lower and hence $x_{BP}$ is also lower to compensate for the lower heat rate. The ORC design point for 57 mph is referenced in Table 4.9. The MOPSO was run for 100 iterations yielding 24 non-dominated solutions.

![Figure 4.19: Pareto optimal points generated from the MOPSO for 57 mph case](image)

Figure 4.19 shows the Pareto Front obtained for the 57 mph case. In the figure, the design point is much closer to the maximum power than in the case of 65 mph. This is due to the fact that the low amount of exhaust heat available leading to lesser excess heat available to sustain higher refrigerant mass flow rates. This can be seen in
Figure 4.20a where point 1 is far away from the boundary of the cuboid pertaining to the pump speed and also in Figure 4.20b where the proximity of point 1 to the upper boundary of refrigerant flow rate is less compared to that of the 65 mph case. The lower available excess heat also means that the performance breadth encompassed by the Pareto front is less than that of 65 mph case. The lowest $P_{\text{net}}$ (point 24) is $\sim 40\%$ of the maximum and the lowest $\dot{E}_d$ (point 1) is $\sim 62\%$ of the maximum which compared to 25% and 40% respectively for the 65 mph case.

![Figure 4.20: Constraints for the 57 mph case.](image)

(a) Control input constraints  (b) System constraints.
### Table 4.9: Design Point at 57mph

<table>
<thead>
<tr>
<th>Vehicle States:</th>
<th>Control Input Positions:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle Speed: 57 mph</td>
<td>Pump Speed: 365 RPM</td>
</tr>
<tr>
<td>Exhaust Temperature: 863 K</td>
<td>Expander Speed: 2933 RPM</td>
</tr>
<tr>
<td>Exhaust Mass Flow Rate: 0.024 kg/s</td>
<td>Exhaust gas bypassed: 16.24 %</td>
</tr>
</tbody>
</table>

**ORC Target States:**

<table>
<thead>
<tr>
<th>Evaporator Outlet Temp: 483 K</th>
<th>Coolant Pump Speed: 4060 RPM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Refrigerant Mass Flow Rate: 0.040 kg/s</td>
<td>Coolant Flow Rate: 104.2 g/s</td>
</tr>
<tr>
<td>Evaporator Pressure: 21 bar</td>
<td></td>
</tr>
<tr>
<td>Condenser Pressure: 2.5 bar</td>
<td></td>
</tr>
</tbody>
</table>

(a) System parameter constraints  
(b) Control input constraints

Figure 4.21: Constraints of the Pareto optimal points
Figure 4.21 shows a clearer representation of the constraints. \( p_H \) and \( T_4 \) follow a trend similar to the 65 mph case but \( m_{ref} \) differs but stopping short well ahead of the constraint due to the lower heat content of the exhaust. \( p_H \) starts to drop after point 13 since the expander speed \( N_e \) has bottomed out. In the next section, a comparison of both the 65 mph case and the 57 mph case will be conducted.

**Comparison between Pareto Front at Design Point and at a Off-Design condition**

![Figure 4.22: Design point and off-design Pareto front comparison normalized with respect to 65 mph extreme points](image)

The Pareto front of the 57 mph case is overlaid with the 65 mph Pareto front as shown in Figure 4.22. Both Pareto fronts have the normalization factor \( P_{ref} \) and \( \dot{E}_{d,ref} \) from the 65 mph case applied. Observe that the 57 mph Pareto has a lower exergy destruction for the same \( P_{net} \) of the 65 mph case. This may be attributed lesser exergy destruction at the exhaust mixing point and evaporator due to the
Table 4.10: Summary of parameters for Pareto optimal points at 57 mph

<table>
<thead>
<tr>
<th>Point</th>
<th>$P_{net}$</th>
<th>$\dot{E}_d$</th>
<th>$x_{BP}$</th>
<th>$N_{fp}$</th>
<th>$N_e$</th>
<th>$p_H$</th>
<th>$T_4$</th>
<th>$\dot{m}_{ref}$</th>
<th>$Q_{evap}$</th>
<th>$\eta_{ORC}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-]</td>
<td>[-]</td>
<td>%</td>
<td>rpm</td>
<td>rpm</td>
<td>bar</td>
<td>K</td>
<td>g/s</td>
<td>[-]</td>
<td>[-]</td>
<td>[-]</td>
</tr>
<tr>
<td>1</td>
<td>1.00</td>
<td>1.00</td>
<td>0.0</td>
<td>415</td>
<td>3051</td>
<td>23.6</td>
<td>500</td>
<td>45.3</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>2</td>
<td>0.93</td>
<td>0.95</td>
<td>9.1</td>
<td>372</td>
<td>2647</td>
<td>24.6</td>
<td>508</td>
<td>40.6</td>
<td>0.90</td>
<td>1.02</td>
</tr>
<tr>
<td>3</td>
<td>0.87</td>
<td>0.94</td>
<td>13.1</td>
<td>359</td>
<td>2815</td>
<td>22.7</td>
<td>505</td>
<td>39.3</td>
<td>0.87</td>
<td>0.99</td>
</tr>
<tr>
<td>4</td>
<td>0.85</td>
<td>0.94</td>
<td>14.4</td>
<td>359</td>
<td>2783</td>
<td>22.5</td>
<td>499</td>
<td>39.3</td>
<td>0.86</td>
<td>0.98</td>
</tr>
<tr>
<td>5</td>
<td>0.82</td>
<td>0.94</td>
<td>17.7</td>
<td>360</td>
<td>2562</td>
<td>22.9</td>
<td>482</td>
<td>39.3</td>
<td>0.84</td>
<td>0.96</td>
</tr>
<tr>
<td>6</td>
<td>0.82</td>
<td>0.92</td>
<td>19.8</td>
<td>354</td>
<td>2227</td>
<td>24.7</td>
<td>480</td>
<td>38.5</td>
<td>0.82</td>
<td>0.98</td>
</tr>
<tr>
<td>7</td>
<td>0.81</td>
<td>0.88</td>
<td>20.7</td>
<td>328</td>
<td>2436</td>
<td>23.6</td>
<td>508</td>
<td>35.9</td>
<td>0.79</td>
<td>1.01</td>
</tr>
<tr>
<td>8</td>
<td>0.78</td>
<td>0.88</td>
<td>23.4</td>
<td>330</td>
<td>2269</td>
<td>23.8</td>
<td>492</td>
<td>35.9</td>
<td>0.78</td>
<td>0.99</td>
</tr>
<tr>
<td>9</td>
<td>0.77</td>
<td>0.85</td>
<td>25.6</td>
<td>309</td>
<td>2091</td>
<td>24.2</td>
<td>508</td>
<td>33.7</td>
<td>0.75</td>
<td>1.02</td>
</tr>
<tr>
<td>10</td>
<td>0.70</td>
<td>0.82</td>
<td>32.1</td>
<td>300</td>
<td>1606</td>
<td>24.6</td>
<td>484</td>
<td>32.7</td>
<td>0.70</td>
<td>0.99</td>
</tr>
<tr>
<td>11</td>
<td>0.68</td>
<td>0.80</td>
<td>33.3</td>
<td>284</td>
<td>1829</td>
<td>23.3</td>
<td>500</td>
<td>31.0</td>
<td>0.68</td>
<td>1.06</td>
</tr>
<tr>
<td>12</td>
<td>0.67</td>
<td>0.78</td>
<td>34.8</td>
<td>277</td>
<td>1669</td>
<td>23.6</td>
<td>502</td>
<td>30.2</td>
<td>0.66</td>
<td>1.01</td>
</tr>
<tr>
<td>13</td>
<td>0.66</td>
<td>0.76</td>
<td>36.2</td>
<td>270</td>
<td>1400</td>
<td>24.6</td>
<td>505</td>
<td>29.3</td>
<td>0.64</td>
<td>1.03</td>
</tr>
<tr>
<td>14</td>
<td>0.64</td>
<td>0.76</td>
<td>38.1</td>
<td>269</td>
<td>1400</td>
<td>24.0</td>
<td>493</td>
<td>29.3</td>
<td>0.63</td>
<td>1.00</td>
</tr>
<tr>
<td>15</td>
<td>0.63</td>
<td>0.75</td>
<td>38.8</td>
<td>262</td>
<td>1400</td>
<td>23.8</td>
<td>499</td>
<td>28.5</td>
<td>0.62</td>
<td>1.01</td>
</tr>
<tr>
<td>16</td>
<td>0.61</td>
<td>0.74</td>
<td>40.6</td>
<td>260</td>
<td>1400</td>
<td>23.2</td>
<td>491</td>
<td>28.3</td>
<td>0.61</td>
<td>0.99</td>
</tr>
<tr>
<td>17</td>
<td>0.60</td>
<td>0.72</td>
<td>41.6</td>
<td>249</td>
<td>1400</td>
<td>22.9</td>
<td>503</td>
<td>27.1</td>
<td>0.59</td>
<td>1.00</td>
</tr>
<tr>
<td>18</td>
<td>0.55</td>
<td>0.69</td>
<td>45.4</td>
<td>237</td>
<td>1400</td>
<td>21.7</td>
<td>495</td>
<td>25.8</td>
<td>0.56</td>
<td>0.97</td>
</tr>
<tr>
<td>19</td>
<td>0.53</td>
<td>0.69</td>
<td>46.2</td>
<td>235</td>
<td>1400</td>
<td>21.4</td>
<td>494</td>
<td>25.5</td>
<td>0.55</td>
<td>0.97</td>
</tr>
<tr>
<td>20</td>
<td>0.52</td>
<td>0.67</td>
<td>47.2</td>
<td>228</td>
<td>1400</td>
<td>21.1</td>
<td>499</td>
<td>24.8</td>
<td>0.54</td>
<td>0.87</td>
</tr>
<tr>
<td>21</td>
<td>0.45</td>
<td>0.63</td>
<td>52.8</td>
<td>213</td>
<td>1400</td>
<td>19.2</td>
<td>483</td>
<td>23.1</td>
<td>0.49</td>
<td>0.92</td>
</tr>
<tr>
<td>22</td>
<td>0.42</td>
<td>0.62</td>
<td>54.7</td>
<td>208</td>
<td>1602</td>
<td>17.7</td>
<td>475</td>
<td>22.6</td>
<td>0.47</td>
<td>0.88</td>
</tr>
<tr>
<td>23</td>
<td>0.39</td>
<td>0.61</td>
<td>57.4</td>
<td>207</td>
<td>1401</td>
<td>17.5</td>
<td>452</td>
<td>22.5</td>
<td>0.46</td>
<td>0.84</td>
</tr>
<tr>
<td>24</td>
<td>0.38</td>
<td>0.61</td>
<td>57.4</td>
<td>205</td>
<td>1671</td>
<td>16.5</td>
<td>458</td>
<td>22.2</td>
<td>0.45</td>
<td>0.83</td>
</tr>
</tbody>
</table>
lower exhaust gas temperature and mass flow rate at 57 mph. Also notice that the location of the weights \( \alpha \) for both the Pareto fronts is approximately at the same exergy destruction rate.

4.6 Conclusion

Efficiency improvements to the combustion process are providing diminishing returns and tightening emissions standards have spurred the development of novel methods to meet consumer and regulatory demands for higher fuel economy. The exhaust of an internal combustion engine releases a significant amount of waste heat and Waste Heat Recovery (WHR) methods are being researched to recycle this waste heat. The Organic Rankine Cycle (ORC) is a promising method of recovering this waste heat. A recuperative ORC was modeled in GT-SUITE with calibration data for the heat exchangers provided by the manufacturer. An operating point for the ORC that provided reasonable performance at 65 mph exhaust conditions was set. This point was deemed as the design point. A methodology to generate feedforward maps for the control inputs to maintain the design point for fluctuating exhaust conditions was devised. Later, a Design of Experiments (DOE) was performed to analyze the behavior of the system with respect to the first and second law of thermodynamics.

Scope for optimization of the cycle was established after finding a trade-off relationship between two parameters, expander power output and exergy destruction. Exergy destruction was proposed to be used as a measure of the distance the operating point is away from the constraints on the control inputs. By expressing the strain on the control inputs through a single parameter, an easier way of formulating a cost function for optimal control may be possible. The Multi Objective Particle Swarm Optimization (MOPSO) algorithm was employed to generate a Pareto front
that would serve to characterize the trade-off relation between the two quantities at the design point exhaust condition of 65 mph.

The MOPSO algorithm returned 22 non-dominated points which form the Pareto front. The Pareto front was then parameterized and an analysis was performed on the non-dominated points after which an operating point was suggested that provided 83% of the maximum power but a safe distance away from the system constraints. The new operating point also provided 50% more power than the design point and increases the cycle efficiency by almost 1%. A Pareto front for an Off-design exhaust condition, 57 mph was performed and compared with the 65 mph case.

The work done in this thesis proves the effectiveness of the PSO algorithm for multi-objective optimization of a black box model of a non-linear thermal system with multiple control variables. The PSO is a relatively efficient algorithm to generate a Pareto front for a multi-dimensional system with little to no information of the system. The algorithm can also achieve major computational advantages through parallelization. Another advantage over other stochastic algorithms is that implementation of constraints on the system is easier.

4.7 Future Work

The solution to a MOO problem is more computationally intensive when compared to a SOO problem. Hence, reformulating the ORC optimization problem into a SOO may be preferable. This may be achieved by parameterizing the Pareto front and selecting a suitable weight $\alpha$ which may be constant or dependent on a set of parameters. If the formulation is possible, then any SOO optimizer can be used to solve the SOO problem for a range of exhaust gas $T_{exh}$ and $\dot{m}_{exh}$ conditions that encompasses the full breadth of exhaust gas during vehicle operation to form a map of optimum performance points for feed forward control implementation.
BIBLIOGRAPHY


153


[34] P. Lamar, “Turbo Compounding The Rotary.” 11


[66] A. Duvia and S. Tavolo, “Application of ORC unit in the pellet Production Field: Technical-Economic Considerations and overview of the operational results of an ORC plant in the industry installed in Mudau(Germany).” 16


[95] “Gamma Technologies GT-SUITE.” 32


[104] “Genetron 245fa Safety Data Sheet,” 2014. 64


164


165


[138] “Rosenbrock Function.” 95


168


