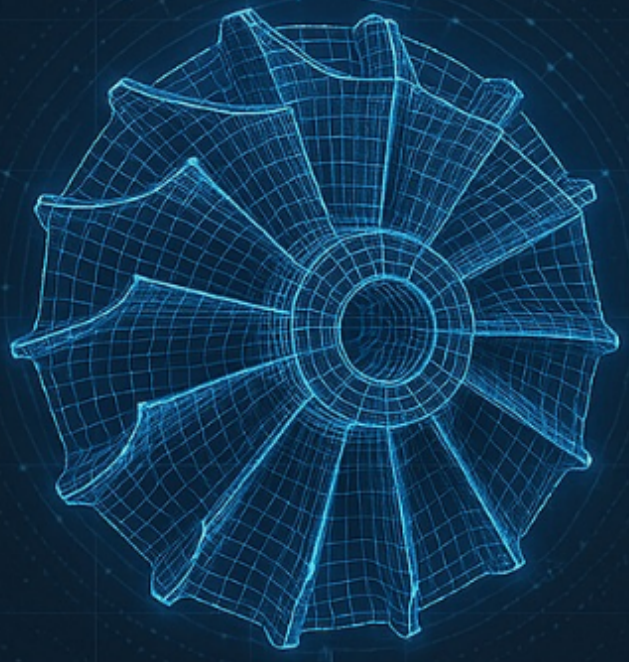
Surrogate Model for Predicting Efficiency and Weight of ORC Turbine for Combined Cycle Engines

Thesis Report

Tota De Hauwere

Delft University of Technology



TU Delft

Surrogate Model for Predicting Efficiency and Weight of ORC Turbine for Combined Cycle Engines

by

Tota De Hauwere

to obtain the degree of Master of Science

at the Delft University of Technology,

to be defended publicly on Thursday April 17, 2025 at 09:30 AM.

Student number:	4551699	
Project duration:	April, 2024 – March, 2025	
Thesis committee:	Dr. F. De Domenico	Chair
	Dr. M. Pini,	Responsible thesis supervisor
	Dr. N. A. K. Doan	External examiner
	Ir. M. Majer,	Daily supervisor

Cover: AI generated
Style: TU Delft Report Style, with modifications by Daan Zwaneveld

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

This page was left blank intentionally.

Preface

This thesis marks the conclusion of my Master's studies in Aerospace Engineering at the Delft University of Technology. It is the result of nearly a year of research, analysis, and writing on the development of a surrogate model using symbolic regression to predict the weight and efficiency of radial inflow turbines for Organic Rankine Cycle (ORC) systems. My background in turbomachinery and thermodynamics, along with my interest in data-driven modeling, motivated me to explore this topic in depth.

The process of writing this thesis has been both challenging and rewarding. A significant challenge was determining appropriate settings for data generation, ensuring that the dataset was both representative and useful for training the surrogate model. Additionally, becoming familiar with TurboSim took considerable time, as understanding the intricacies of the program and its functionalities was essential for generating reliable simulation data. Overcoming these challenges has helped me grow both technically and personally. I especially enjoyed working on the implementation of symbolic regression techniques and analyzing their impact on turbine performance modeling.

I would like to thank my supervisors, Dr. Matteo Pini and especially Ir. Matteo Majer, for their invaluable guidance, insightful feedback, and continuous support throughout this thesis project. Their expertise and encouragement have been crucial in shaping this research. Additionally, I acknowledge the Delft Blue supercomputer for providing the computational resources necessary for this research.

A special thank you to my friends and family, whose support, encouragement, and motivation have made this journey much more enjoyable. Whether through discussions, feedback, or simply sharing the challenges of research, their presence has been invaluable. My family's consistent belief in me has been a constant source of strength throughout my academic journey, and I am deeply grateful for their patience and encouragement.

Finally, I hope that this thesis contributes to the ongoing research in surrogate modeling for ORC turbines and provides a useful reference for future studies in this field.

Thank you for taking the time to read my work. I hope you find it insightful.

Tota De Hauwere
Delft, April 2025

Summary

Organic Rankine cycle (ORC) turbines for combined cycle engines offer significant potential for improving fuel efficiency. However, turbine weight remains a critical design factor, as the turbogenerator can account for up to one-third of the power unit mass [28]. The turbine design tool TurboSim, developed by Majer and Pini [34], is computationally expensive. This study aims to develop an accurate and computationally efficient symbolic regression surrogate model to predict ORC radial-inflow turbine (RIT) efficiency and weight. Additionally, it examines the impact of working fluid selection and key geometrical parameters on turbine efficiency and weight.

A parametric study was performed to identify the six design variables (DVs) that have the highest influence on net total-to-total efficiency ($\eta_{tt_{net}}$) and turbine weight (W_{turb}), which includes the stator blades, impeller, backplate and shroud casings, as well as a locking ring. The training data is generated using TurboSim [34] and split into a 90% training set and a 10% test set. The surrogate models are trained on working fluid, indicated by the molecular complexity; mass flow rate; volumetric flow ratio and compressibility factor, as well as the six most influential DVs on efficiency and weight, which are the following: R_3/R_2, R_h/R_t, $L_{ax}/\Delta R$, $\phi_{2,is}$, ψ_{is} and $(g/h)_{le}$. The model performance was evaluated using mean-squared-error (MSE) and R-squared (R^2). The MSE gives an indication of the model accuracy, while the R^2 indicates how well the trend in the data is captured. Using both of these metrics allows for comparing the performance of the different models.

While a single multi-output model was initially explored, separate models for each working fluid provided better accuracy. Four fluid categories — refrigerants (R134a), hydrocarbons (butane, cyclopentane, toluene), alcohols (ethanol), and siloxanes (MM) — were analyzed, showing that working fluid has a minor effect on efficiency but a significant impact on turbine weight. A parametric study identified key design variables, with impeller radius ratio (R_3/R_2) and hub-to-tip radius ratio (R_h/R_t) having the greatest impact. While efficiency predictions were generally reliable, weight predictions exhibited greater variability, with lower R^2 values and higher MSE for some fluids.

This research provides an efficient method for estimating ORC turbine performance, reducing computational costs in early-stage design. The insights on parameter influence can aid engineers in making informed decisions, potentially leading to improved turbine designs and broader adoption of ORC technology.

The results highlight the potential of symbolic regression for rapid and interpretable turbine design optimization. However, further research is needed to improve weight prediction consistency.

Contents

List of Figures

List of Tables

Nomenclature

Abbreviations

Abbreviation	Definition
AEA	All Electric Aircraft
ANN	Artificial Neural Network
APU	Auxiliary Power Unit
BPR	Bypass Pressure Ratio
CC	Combined Cycle
DV	Design Variable
DNN	Deep Neural Network
EA	Evolutionary Algorithm
FFNN	Feed Forward Neural Network
GA	Genetic Algorithm
GHG	Greenhouse Gas
GP	Genetic Programming
LB	Lower Bound
MEA	More Electric Aircraft
NN	Neural Network
OPR	Overall Pressure Ratio
ORC	Organic Rankine Cycle
PFC	Perfluorochemical
RISE	Revolutionary Innovation for Sustainable Engines
RIT	Radial-Inflow Turbine
RNN	Recurrent Neural Network
SAF	Sustainable Aviation Fuel
SR	Symbolic Regression
SR1E	Symbolic Regression trained for Efficiency
SR1W	Symbolic Regression trained for Weight
SR2O	Symbolic Regression 2 Outputs
SWITCH	Sustainable Water-Injecting Turbofan Comprising Hybrid-Electrics
TIT	Turbine Inlet Temperature
TS	Turboshaft
UB	Upper Bound
WET	Water Enhanced Turbofan

Symbols

Symbol	Definition	Unit
g	Gap	[m]
H	Blade height	[m]
l	Length	[m]
L	Length	[m]
$\dot{m}$	Mass flow rate	[kg/s]
N	Molecular complexity	[-]
R	Radius	[m]
t	Thickness	[m]

Symbol	Definition	Unit
U	Velocity	[m/s]
V	Volume	[kg/m^3]
VR	Volumetric flow ratio	[-]
W	Weight	[kg]
Z	Compressibility factor	[-]
Z	Number of blades	[-]
α	Flow angle	[°]
Δ	Difference	[-]
η	Efficiency	[%]
ϕ	Flow coefficient	[-]
ψ	Head coefficient	[-]
ρ	Density	[kg/m^3]

Subscripts

Subscript	Definition
0	Stator inlet
1	Stator outlet
2	Inducer or rotor inlet
3	Exducer or rotor outlet
ax	Axial
bf	Backface
bp	Back plate
$cond$	Condensation
h	Hub
is	Isentropic
le	Leading edge
m	Meridional
r	Reduced
rot	Rotor
st	Stator
t	Tip
t	Total
te	Trailing edge
tt	Total-to-total

1

Introduction

The aviation industry has undergone a process of progressive decarbonization throughout the past decades. The general trends are 1) improving fuel efficiency thanks to an increased thermodynamic efficiency due to increasing the maximum cycle temperatures and overall pressure ratio, and 2) an increased propulsive efficiency by increasing the engine bypass ratio. The industry is now also turning towards more electric and all electric aircraft. One way of achieving further improvement is to use combined cycle configurations which can use the unused thermal power available at the exhaust of turbine engines.

Organic Rankine cycle (ORC) combined cycle (CC) systems show very promising results in increasing fuel efficiency. A study performed by Majer and Pini [34] showed that the turbine stage efficiency of high-pressure ratio supersonic radial-inflow turbine (RIT) of ORC systems can be predicted independently of the working fluid. However, an important objective as weight was not considered in the work. Weight is an important parameter to consider because the turbogenerator of an ORC system can take up one-third of the power unit mass [28]. This highlights the need to further investigate how such a turbogenerator can be made lighter by increasing the amount of detail in the turbogenerator design optimization. Another concern that the industry might have is that performing these optimizations can be very time-consuming. A solution to this problem is to use machine learning to create a surrogate model that can predict efficiency and weight of radial-inflow turbines at a very low computational cost. The aim of this research is to propose a computationally efficient and highly accurate method to predict the efficiency and weight of ORC radial-inflow turbines for combined cycle engines.

The report is structured as follows: the motivation for the study, which includes background information on the problem and key findings from the literature review on various machine learning algorithms, is presented in chapter 2. Chapter 3 outlines the project description, including the research objective and the research questions. Chapter 4 details the research methodology, explaining the approach used to calculate turbine weight and the parametric study conducted to identify the design variables with the greatest influence on efficiency and weight. Additionally, the development of the surrogate model is discussed. Chapter 5 presents and analyzes the results of data generation and surrogate modeling. Finally, chapter 6 provides the conclusions and limitations of the study, and recommendations for future research.

2

Background

2.1. Motivation of Study

According to the Paris Agreement stipulated in 2015, the global average temperature rise should be kept well below 2°C above pre-industrial levels to mitigate the negative effects and limit the consequences of the ongoing climate change [36]. Among the several contributing factors, lowering the global amount of greenhouse gas (GHG) emissions emitted in the atmosphere is of primary importance.
The European Commission has created the European Green Deal[1] to be able to reach the commitments stated in the Paris Agreement. One of the initiatives stated in the European Green Deal is 'Fit for 55'[2]. The target for this package is to reduce net greenhouse gas emissions by at least 55% by 2030.

The aviation industry, with a share of about 2–2.5% of the global CO_2 emissions[3] [17], has undergone a process of progressive decarbonization throughout the past decades. To cite the main ones, engines for commercial aircraft saw a rapid increase in fuel efficiency thanks to, on the one hand, ever-increasing maximum cycle temperatures (TIT) and overall pressure ratio (OPR) to boost the engine thermodynamic efficiency. On the other hand, propulsive efficiency has steadily increased by increasing the engine bypass ratio (BPR) [10, 21, 31]. From 2005 to 2019, the fuel efficiency has improved by 39%, however, this cannot offset the large absolute growth of emissions[3]. CO_2 is not the only emission that is produced by flying. There are also non-CO_2 warming effects, e.g., condensation of water in contrails at high altitudes. These can lead to a net increased warming effect. Another chemical process that happens at high altitudes is the production of ozone from NOx, which is a greenhouse gas as well [17]. To be able to reach the previously mentioned goals, the industry will need to investigate disruptive solutions, such as alternative fuels and other non-combustion based technologies.

Airbus announced the ZEROe[4] project, where they are developing three hybrid-hydrogen powered and one electric aircraft by 2035. Faber and Lee [13] state that fully electric aircraft will be limited to shorter ranges since batteries are heavy. Thus, the contribution to the reduction of emissions will only have a small effect on the larger picture. A bigger impact could be achieved when medium and long range aircraft are converted into more electric aircraft (MEA) or all electric aircraft (AEA). However, the electrification of an aircraft increases the complexity of the power systems on board of it [4]. The following three alternative fuel propulsion related projects aim to reduce fuel consumption as well as CO_2 emissions: RISE[5], SWITCH[6] and UltraFan[7]. The CFM Revolutionary Innovation for Sustainable

[1] https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal_en, accessed on April 3rd 2024

[2] https://www.consilium.europa.eu/en/policies/green-deal/fit-for-55-the-eu-plan-for-a-green-transition/, accessed on April 3rd 2024

[3] https://www.mckinsey.com/industries/aerospace-and-defense/our-insights/decarbonizing-aviation-executing-on-net-zero-goals#/, accessed on April 3rd 2024

[4] https://www.airbus.com/en/innovation/low-carbon-aviation/hydrogen/zeroe, accessed on April 3rd 2024

[5] https://www.safran-group.com/videos/what-rise-program-sustainable-engines, accessed on April 16th 2024

[6] https://www.airbus.com/en/newsroom/press-releases/2022-11-clean-aviation-switch-project-to-advance-hybrid-electric-and-water, accessed on April 16th 2024

[7] https://www.rolls-royce.com/innovation/ultrafan.aspx, accessed on April 16th 2024

Engines (RISE) project is a joint venture between Safran and General Electric. They are developing an open fan engine[8] that is fully compatible with alternative fuels. The RISE engine is expected to have a 20% CO_2 emissions and fuel consumption reduction [22]. The SWITCH (Sustainable Water-Injecting Turbofan Comprising Hybrid-Electrics) project combines Water Enhanced Turbofan (WET) and hybrid-electric propulsion. The aim is to reduce CO_2 emissions and fuel consumption by 25% compared to today's state-of-the-art engines[6]. Rolls-Royce created the UltraFan, which is a demonstrator engine that can operate on 100% Sustainable Aviation Fuels (SAF) and is designed to have 40% less NOx, 25% decrease in fuel consumption and a 35% decrease in noise at cruise[7].

Hybrid aircraft are also an option to be able to transition to fully electric alternatives. The authors of [37] highlighted different technological solutions and the challenges ahead of MEA. In this context, researchers of TU Delft [29] looked into hybridizing the Auxiliary Power Unit (APU) using combined cycle architectures. The APU provides secondary power for the main engine start-up and electricity for all the electrical components on the aircraft.

Combined cycle configurations can be interesting to look at because of the large amount of unused thermal power available at the exhaust of gas turbine engines. By combining multiple thermodynamic cycles in a topping (high temperature) and bottoming (low temperature) cycle, one can use the rejected heat from the topping cycle in the bottoming cycle. This can lead to increased efficiency and power output [24] depending on the selected ORC working fluid, the size and performance of the heat exchangers, and the isentropic efficiency of the ORC turbine. Krempus et al. [29] showed that thermal power harvesting from the exhaust gases of a modern day APU can lead to efficiency improvements in the range of 1% of mission total fuel mass. A combined cycle APU (CC-APU) system was designed and optimized numerically, featuring an ORC system as a bottoming cycle of the Joule-Brayton cycle. However, since the aircraft APU is usually operated only during taxiing and for on-ground power, the potential for fuel savings resulting from APU efficiency improvements is rather limited.

Later research by Krempus et al. [28] focused on the preliminary design and assessment of a combined cycle turboshaft (CC-TS) engine using an ORC as a bottoming cycle to recover exhaust gas thermal power. Such a combined cycle engine would then drive electrical generators that provide the power needed by the turboelectric propulsion system. A multidisciplinary optimization (MDO) framework including models for the engine, ORC system, ORC turbine, heat exchangers, and mission analysis was created for this purpose. The results presented that a fuel saving of 4% could be reached when using the optimized system instead of the aircraft employing turboshaft engines. To maintain the computational cost within reasonable limits, the number of design variables used in the optimization was limited to 18 and mainly included system design variables such as the OPR, maximum ORC temperature, maximum ORC pressure, et cetera. The ORC turbogenerator was designed including a radial-inflow turbine stage driving a high-speed permanent magnet generator [28, 29], and it was optimized by including only three design variables, namely the work coefficient, the flow coefficient and the impeller radius ratio, related to the RIT stage. Nevertheless, the optimization of a radial turbine for ORC applications encompass several other design variables as shown by other authors [11, 35]. An in-house sensitivity study performed by Majer shows that fuel consumption can be further improved by looking into different working fluids, size and configurations (ORC turbogenerator vs direct-drive), showing that optimizing an ORC turbine is desirable.

Majer and Pini [34] studied the design guidelines for high-pressure ratio supersonic RIT of ORC systems. A reduced-order model of RIT including a loss model based on the first principles is presented. In the investigation, it was shown that one can find a simplified turbomachinery equation to predict stage efficiency independently of the working fluid. The model cannot be generalized to optimize a turbogenerator (including volute, diffuser and generator), however, it gives a preliminary prediction to guide the design based on the parameters given. The authors managed to achieve further order reduction by means of simple data fitting to predict the efficiency of the ORC turbine, but other important objectives such as turbine weight were not considered in the work.

[8] An open fan engine is different from a turboprop engine and it is able to fly at $M = 0.8$, similar to current single aisle aircraft

The weight breakdown of the ORC system for the optimized CC power unit (CC-PU) configuration found in [28], illustrated in Figure 2.1, shows that one-third of the power unit mass (total weight excluding generator (m_{gen}) and turboshaft (m_{ts})) is taken up by the turbogenerator (m_{tg}). This highlights the need to further investigate how such a turbogenerator can be made lighter by increasing the amount of detail in the turbogenerator design optimization. The process flow diagram of the combined cycle turboshaft proposed in this report is shown in Figure 2.2.

Giuffré et al. [15] presented a data-driven model for high-speed centrifugal compressors for aircraft environmental control systems. They showed that by using an Artificial Neural Network (ANN) surrogate model (Multilayer Perceptron), the total computational cost can be reduced by a factor of 3.33 (125h to 27.5h). By using a filtered dataset, the training computational cost was reduced by 30% without any loss of useful information. ANN require hyperparameter optimization, however, other types of machine learning algorithms might be a better choice for the surrogate model. Exploring whether a similar approach can be applied to a turbine model can provide valuable insights into computational efficiency and model performance.

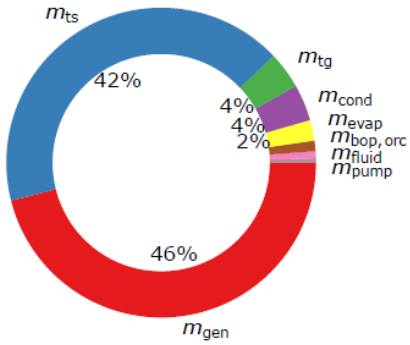


Figure 2.1: CC-TS mass breakdown. The dry mass of the engine without generators m_{ts} is shown in blue, the generator mass m_{gen} is shown in red and the ORC turbogenerator mass m_{tg} is shown in green. One can see that m_{tg} is about one third of the left over mass (total - engine - generator) [28]

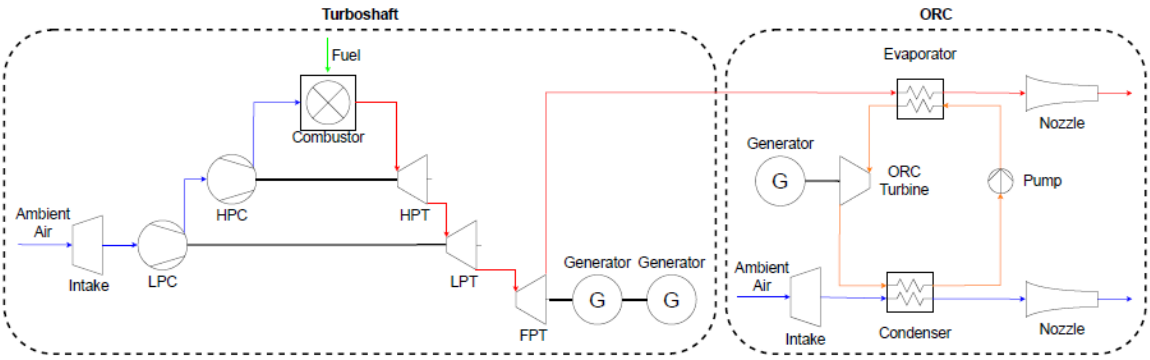


Figure 2.2: Process flow diagram of the Combined Cycle Turboshaft configuration discussed by Krempus et al. [28]

2.2. Algorithm selection

A literature study was conducted before starting the thesis project. A more detailed version was written for the course AE4020 Literature Study, however, the key findings are provided in the following sections.

Many machine learning algorithms have been developed over the decades. The following section provides a brief overview of those that are suitable for developing a surrogate model for ORC turbine design. The Literature Study report, as mentioned previously, also discusses some algorithms that are not suitable; however, they have been omitted from this report for the sake of conciseness.

A set of requirements was defined to filter through the large number of algorithms and determining which ones best meet the specified criteria. The requirements are as follows:

- *Can use labeled datasets:* It is preferred that the algorithm can use labeled datasets, because the training data will be structured in labeled format. However, it is not a hard requirement. If an algorithm would outperform the others and label the data in its own structured way, that is accepted as well.
- *Able to handle large datasets well:* This surrogate is created to predict the efficiency and weight of ORC turbines. To be able to have an accurate prediction for a wide variety of input parameters, the surrogate model must be trained with a very large number (> $100\,000$) of unique turbine configurations.
- *Can predict nonlinear relationships:* The chance of finding an accurate linear relationship is very small. To be certain that the model can make an accurate prediction, the algorithm used should also allow for nonlinear relationships. This will probably apply to most algorithms that are investigated; however, it should be mentioned.
- *Efficient algorithm:* A key objective of this study is to develop a computationally efficient model by minimizing computational time. Therefore, a fast algorithm is preferred. It is important to note that while the training process may be time-consuming, it is a one-time effort required solely for developing the surrogate model.

Two types of algorithms were selected as suitable candidates for developing the surrogate model: neural networks and evolutionary algorithms, which are discussed in subsection 2.2.1 and 2.2.2, respectively.

2.2.1. Neural network

Neural networks (NN) were initially developed to adress problems that could be solved using linear regression[9]. They are specifically designed to efficiently identify low-dimensional patterns in high-dimensional data [9].

Artificial Neural Networks (ANNs) are inspired by the interconnection of neuron cells in a brain. An ANN has three types of layers: input, hidden and output layer, which is very similar to the parts in a neuron: dendrites (receptor), soma (processor of electric signal), nucleus (core of neuron) and axon (transmitting end of neuron) [2].

The advantages and disadvantages of NN are summarized below.

Advantages [19]:

- Automatic feature extraction
- Modeling of non-linear dependencies
- Various architectures for supervised and unsupervised tasks
- Transfer learning
- Flexible
- One hidden layer is enough to approximate any continuously differentiable function

Disadvantages [19]:

- Computationally expensive
- Huge amount of training data needed
- Sensitive to initial randomization of weight matrix
- Can get stuck in local minima
- Each type of NN is designed to solve one class of problems

Leijnen and van Veen [32] have created the Neural Network Zoo, where they give an overview of commonly used neural network architectures. Four types will be discussed in this report, namely multilayer perceptron, feed-forward, recurrent and deep neural networks. A simplified visual representation of these networks are shown in Figure 2.3.

[9] `https://www.baeldung.com/cs/neural-net-advantages-disadvantages`, accessed on March 15th 2024

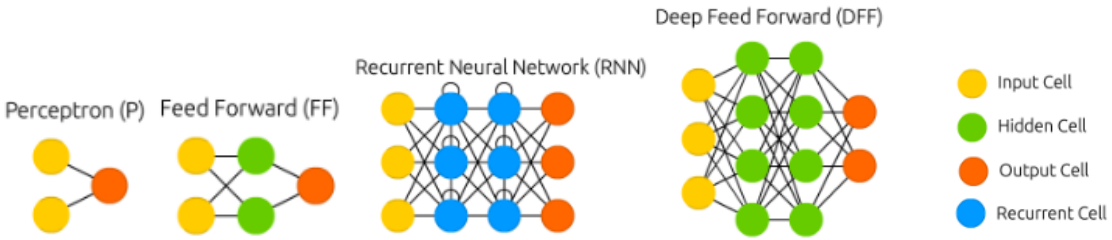


Figure 2.3: Neural network types [32]

Multilayer Perceptron

The perceptron, introduced by Frank Rosenblatt in 1958, is considered the foundation of Artificial Neural Networks, which are based on neuron interaction in the brain [2, 12]. A perceptron is the simplest type of NN, as it does not contain any hidden layers. However, most modern NNs are more complex and use hidden layers in their NN structure. A Multilayer perceptron (MLP) (used by Giuffré et al. [15]) consists of multiple layers of perceptrons, as the name suggests and it will use step-function nonlinearities in the hidden units [6].

Feed-forward Neural Network

A feed forward neural network (FFNN) will, as the name implies, feed the information forward from input to output. Connections can be formed between different layers, however, they cannot form feedback loops (see recurrent neural network) [7]. A single layer will not have any connections, while two adjacent layers will be fully connected. It has information about ”what goes in” and ”what we want to have coming out” [32].

A neural network differs from a multilayer perceptron because the NN uses continuous sigmoidal nonlinearities in the hidden units, while the perceptron uses step-function nonlinearities. Thus, the functions in the NN are differentiable with respect to the network parameters. This will have a major role when training the network [6]. If the NN would have linear activation functions for all hidden units, then there would exist an equivalent network without the hidden units. This is due to the fact that the composition of successive linear transformations is itself a linear transformation [6].

Recurrent Neural Network

A recurrent neural network (RNN) is dependent on the order you feed the input to the network and how it is trained [32]. RNNs take information from downstream nodes and feed the information back into previous nodes, creating feedback loops. These feedback loops can allow you to have memory and model, learn and memorize time series trends. For systems that evolve in time, RNN would be a very suited option.

Deep Neural Network

Deep neural networks (DNNs) are more evolved than ANNs. They usually consist of semi-supervised learning algorithms and many hidden layers, which are fully connected [23]. Deep NN are created to handle very large and complicated problems. However, DNNs are very complex, difficult to scale and have high computational costs [12, 23].

2.2.2. Evolutionary Algorithms

Evolutionary learning is inspired by biological organisms that can adapt to the environment [2].The algorithm analyzes the system’s behavior and adjusts based on the inputs, eliminating less probable solutions. It operates on the principle of fitness to identify the most optimal solution for the problem.

Any evolutionary algorithm will have the following six steps [9]:

1. A population of individual is defined, along with a fitness function and a set of mutation operators.
2. A randomly chosen subset of size n_s is selected from the population. n_s usually equals 2, however larger numbers are allowed.

3. Each individual in this subset is evaluated based on the fitness function.
4. The fittest individual is chosen as the winner with probability p; if not selected, it is removed from the subset, and the process repeats until only one remains, which is then selected.
5. A duplicate of the chosen individual is created and a randomly selected mutation is applied.
6. The mutated individual replaces an existing member of the population.

Genetic Programming (GP) is an Evolutionary Algorithm (EA) technique developed by Koza (1992) [27] and it is similar to Genetic Algorithms (GA), however, they do differ in some parts. Both are based on evolution, however, GA evolves solutions, while GP evolves a program. The most common application of GP is symbolic regression. GP approaches building surrogate models by searching for the best functional combination of basis functions. This differs from other approaches where they will try to find the best linear combination of the basis functions [14]. GP models are directly created from the datasets given as they do not require a specific model to learn the mathematical expression needed for the basis functions.

Genetic Programming uses a tree-like structure where each interior node of an expression tree is occupied by an arithmetic operation from a function set (e.g. sum, division, square root, etc.). The number of children of an interior node must match its function arity (i.e., the number of operands of the operation). Three methods can be used for initializing the population [14]:

1. *Full method*: As the name implies, this method will produce full trees with a specified maximum depth.
2. *Grown method*: It will create trees with different shapes and sizes. It will select nodes randomly until the specified maximum depth is reached.
3. *Ramped half-and-half method*: As suggested by the name, it will create a tree by using the previously mentioned methods to crease each half of a tree.

Every iteration consists of two parts, namely the selection and evolution operators. During the selection stage, the reproduction solutions are selected based on their fitness value. During the evolution operators stage, new solutions are created using crossover, mutation and reproduction operators in a probabilistic way to select new solutions. After performing these two parts, a new population is formed [14, 20].

Crossover describes the process when two nodes are randomly selected, for each parent, and the subtree of the first parent is replaced with the subtree of the second parent. The mutation operator will take a solution and mutate or randomly modify it. This solution can be either a subtree mutation or a node mutation. Lastly, the reproduction operator will reproduce or copy the solution and pass it on to the next generation.

Symbolic Regression

One of the problems commonly used with GP is symbolic regression (SR). Symbolic regression [27] can be classified as a supervised learning algorithm which will search the space of mathematical expressions, while minimizing various error metrics and model complexity [9, 38]. An implicit function is typically expressed as $f(\vec{x}, y) = 0$, whereas an explicit function takes the form $y = f(\vec{x})$. Implicit equations are often more versatile and can effectively describe complex surfaces or multi-output functions in a compact manner. In SR, the objective is to automatically identify implicit equations that best represent experimental data. Instead of traditional SR, which predicts a specific signal or value, implicit equations are formulated to always evaluate to zero across the dataset, ensuring consistency in their representation.

SR problems can also be solved by using a different approach, e.g. deep learning methods. These methods are dependent on pre-defined models and the training dataset to solve the given SR problem, while GP will focus on finding the solution in a search space [20]. Symbolic regression is used to fit a model to a data set, and it is based on the use of expression trees [14].

A recent development in symbolic regression is its application in interpreting neural networks. Cranmer et al. [9] developed PySR, a Python package that applies evolutionary algorithms to discover symbolic equations. While it is primarily designed for symbolic regression on raw data, it can also be integrated

with neural networks to extract interpretable mathematical expressions that approximate the network's behavior.

PySR

PySR is an open-source library for practical symbolic regression developed by M. Cranmer [9]. It was designed to address limitations found in many existing algorithms, making them more applicable in a broader setting. The package integrates a Python interface with a Julia backend, which utilizes just-in-time compilation to enhance performance.

The internal search algorithm used by PySR is a multi-population evolutionary algorithm designed to optimize unknown scalar constants in newly discovered empirical expressions. Parallelization is used to speed up training time required. By using a migration step in the outer loop of PySR that allows to migrate on a global permanent hall of fame, the computational search time can be sped up significantly, which in return decreases the computational cost [9]. The inner and outer loops of PySR are shown in Figures 2.4 and 2.5, respectively.

PySR has very attractive features, which are presented below and a full overview of the software's abilities presented in [9]:

- *Noisy data*: It can work with noisy data because it has a denoising preprocessing step that uses the kernel stated in Equation 2.1 to optimize the Gaussian process.

$$k(x, x') = \sigma^2 \exp\left(\frac{-|x - x'|^2}{2l^2}\right) + \alpha\delta(x - x') + C \tag{2.1}$$

- *Custom losses*: It can work with any user-defined loss function.
- *Custom operators*: If the operator can de defined as either $f : \mathcal{R} \longrightarrow \mathcal{R}$ or $\mathcal{R}^2 \longrightarrow \mathcal{R}$, it can be used in PySR.
- *Feature selection*: PySR uses a gradient-boosting tree algorithm to first fit the dataset, and then select the most important features.
- *Constraints*: Hard constraints can be specified and will be enforced at every mutation.

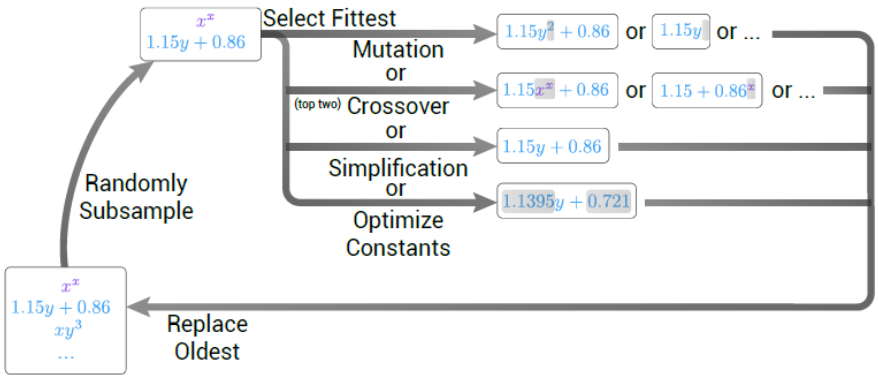


Figure 2.4: Inner loop of PySR showing the evolutionary operators [9]

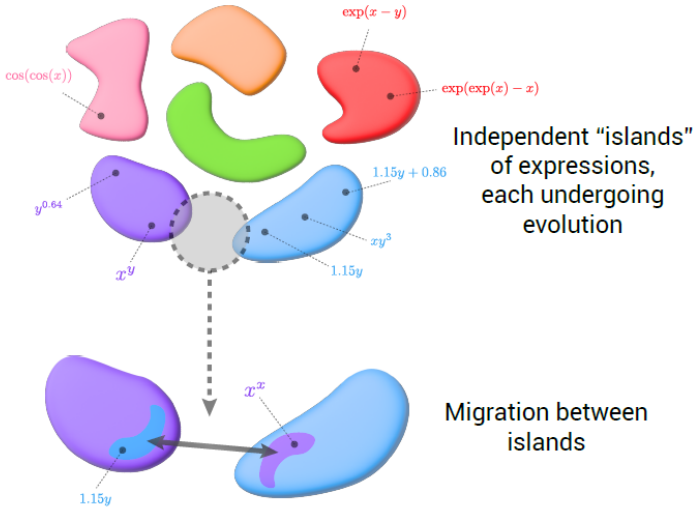


Figure 2.5: Outer loop of PySR showing the migration process between islands [9]

3

Project Description

3.1. Problem Statement

As discussed in chapter 2, modeling ORC enhanced combined-cycle turboshaft (CC-TS) engines for More Electric Aircraft (MEA) entails a large number of design variables. Consequently, component modeling, such as that of the ORC turbogenerator, must be as simple and computationally efficient as possible. Due to the numerous constraints and limitations of size and power-to-weight ratio onboard of an aircraft, investigating the design of an ORC turbogenerator for the CC-TS requires a flexible tool that can be used to study different configurations. To address these challenges, the following research objective was formulated:

Research Objective

This research aims to propose a computationally efficient and highly accurate method to predict the efficiency and weight of ORC radial-inflow turbines.

3.2. Research Questions

As already mentioned in the Research Objective stated above, the aim of the project is to propose a computationally efficient and highly accurate method to predict efficiency and weight of ORC RIT. A surrogate model must be created to be able to comply with the computationally efficient requirement. As demonstrated by Giuffré et al. [15], surrogate modeling can significantly reduce computational time. While Artificial Neural Networks (ANN) are commonly used for this purpose, other algorithms, such as symbolic regression (SR), may also be capable of generating fast and accurate surrogate models. This leads to the first research question of the thesis:

Research Question 1

Can a symbolic regression surrogate model be developed for predicting the efficiency and weight for ORC RIT?

The choice of working fluid has a significant impact on the efficiency and overall performance of ORC systems [24, 25]. Additionally, the state (wet or dry) of the fluid has an impact on the turbine blade life [1]. Since different working fluids operate under different conditions, they can lead to changes in the turbine architecture, which may in return have a considerable impact on turbine weight. Given these factors, the working fluid is a crucial parameter to consider when developing the surrogate model. Therefore, the second research question will examine the impact of working fluid on ORC efficiency and weight.

Research Question 2

What is the impact of working fluid on ORC RIT efficiency and weight?

The design process of a turbine is often complex and time-consuming. It involves multiple factors and parameters, each influencing the final performance. These parameters, known as design variables (DVs), include aspects such as geometry, material properties, and operating conditions, all of which can have varying degrees of impact on the turbine's efficiency and weight. Given the large number of design variables, it is essential to identify which ones have the most significant role in determining these critical performance metrics. This leads to the formulation of Research question 3, presented below.

Research Question 3

What are the design variables that have the largest impact on efficiency and weight?

Lastly, the accuracy of the surrogate model is a critical aspect to consider, as it directly impacts the reliability and effectiveness of the model predictions. Given the complexity of the turbine design process and the variety of fluids that can be used, it is important to assess how accurately the surrogate model can predict turbine efficiency and weight. Furthermore, since different working fluids exhibit varying thermodynamic properties, it is crucial to determine if the accuracy of the surrogate model is consistent across different fluids. Evaluating its performance under various operating conditions will provide insights into the model's generalizability and highlight any potential limitations.

Research Question 4

What is the accuracy of the surrogate model and will it change with working fluid?

4

Methodology

The following chapter will cover the methodology applied to create the surrogate model for predicting efficiency and weight of organic Rankin cycle (ORC) turbines for combined-cycle (CC) engines. First, the method developed to determine the turbine weight will be explained in section 4.1. Then, a parametric study was performed to identify the parameters that have the largest impact on net total-to-total efficiency and turbine weight, which is explained in section 4.2. Lastly, the development of a surrogate model for radial-inflow turbines is discussed in section 4.3, with emphasis on the available model architectures, the training setups, and verification of the resulting surrogate.

4.1. Turbine Weight Estimation

The method developed to estimate the radial-inflow turbine weight will be discussed in this section. A part of the objective of this work is to show the abilities of a symbolic regression in predicting turbine weight. The following assumptions were made:

1. A simple geometry and volume estimation method provides a sufficiently accurate turbine weight to show the abilities of a symbolic regression surrogate model to predict turbine weight accurately.
2. The casing thickness is fixed for all turbine designs, as well as the length of the outer part of the backplate casing, shown in Figure 4.1.

Disk integration or disk method calculates the volume of a solid. For sections of the turbine with a constant inner and outer radius, the volume can be calculated as follows:

$$V = \pi \left(R_2^2 - R_1^2\right) l, \tag{4.1}$$

where V is the volume, R the radii ($R_2 > R_1$) and l is the length of the 3D shape.
For sections where the radius varies along the length of the turbine, the volume is determined using Equation 4.2:

$$V = \int_{x_1}^{x_2} \pi \left(R_2(x)^2 - R_1(x)^2\right) dx, \tag{4.2}$$

where $R_2(x)$ and $R_1(x)$ are functions of the axial coordinate x. This approach ensures an accurate volume estimation for both constant and varying radius sections of the radial-inflow turbine.

Figure 4.1 shows the radial-inflow turbine in a *meridional view*[1]. It contains the impeller *blisk*[2], stator blades and casings (backplate and shroud). To be able to have an accurate estimation of the weight using the disk method, the turbine and its casings need to be divided into multiple parts (indicated in red). The backplate casing (part 1) is divided into two parts (A and B) to estimate the volume correctly when applying Equation 4.1. The stator blades are assumed to take up 50% of the volume of the nozzle

[1] The *meridional view* is a cross-section of the turbine along the axis of rotation.
[2] Radial turbine impellers are typically referred to as *blisks*, which stands for *bladed disk*, and indicates a disk and blades in a single solid piece.

(part 2). Part 3 and 4 make up the impeller blisk and the shroud casing is shown in part 5. The shaft is indicated by a blue hatching pattern and is excluded from the turbine weight. The length from the start of backplate casing to the stator blades was fixed to 10 mm, the thickness of the backface and shroud casing were set to 0.5 mm and 5 mm, respectively. The contour of the rotor blades was determined based on assumptions presented by Glassman [16]. He stated that the contour of the rotor blade hub can be approximated by a quarter-ellipse, when drawn in the meridional view, and the tip contour by a circle. However, a discrepancy was observed when comparing the contours in TurboSim using spline integration and the ellipse approximation. Adjusting the tip contour to a quarter-ellipse resolved this discrepancy, as shown in Figure 4.2.

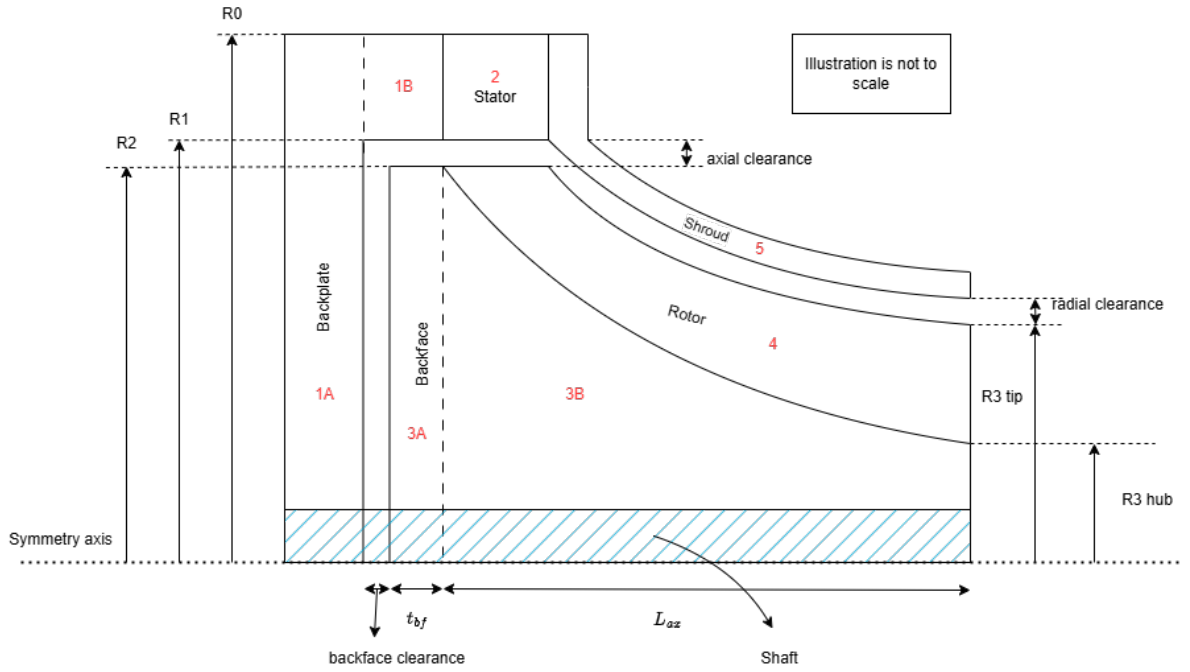


Figure 4.1: Sketch of radial-inflow turbine

The following section explains how the quarter-ellipses are calculated. The center of both ellipses is located at $(x, y)_{center} = (0.01 + L_{ax}, R_2)$. The start and end point of the impeller hub contour are $(x, y)_{inducer,hub} = (0.01, R_2)$ and $(x, y)_{exducer,hub} = (0.01 + L_{ax}, R_{3,hub})$, Similarly, the tip contour starts at $(x, y)_{inducer,tip} = (0.01 + z_{0,tip}, R_2)$ and ends at $(x, y)_{exducer} = (L_{ax}, R_{3,tip})$. An ellipse has two types of axis, namely the major (largest) and minor axis (smallest). The orientation of an ellipse is determined based on the orientation of the major axis. When the horizontal difference $(x_{end} - x_{start})$ is larger than the vertical difference $(y_{end} - y_{start})$, then the ellipse has an horizontal orientation, meaning that the major axis is in a horizontal position. The opposite is true for cases that have its major in a vertical position. The elliptic curve can be found by applying the formulae below:

Horizontal orientation:

$$x_{ellipse} = x_{centre} + \Delta x \cdot \cos(\theta) \quad (4.3)$$

$$y_{ellipse} = y_{centre} + \Delta y \cdot \sin(\theta) \quad (4.4)$$

Vertical orientation:

$$x_{ellipse} = x_{centre} + \Delta x \cdot \cos(\theta) \quad (4.5)$$

$$y_{ellipse} = y_{centre} + \Delta y \cdot \sin(\theta) \quad (4.6)$$

where Δx and Δy are the semi axes of the ellipse and θ are the locations of the third quadrant of the ellipse in radians.

The contour of the shroud is also calculated using the ellipse approximation to ensure compliance with the specified radial clearance between the rotor blades and the shroud casing. The meridional channel contours, from the blade hub to shroud casing, are shown in Figure 4.2. It is clear that the shroud contour aligns perfectly with the spline, which was initially used in TurboSim, confirming the earlier

statement that the all contours could be calculated using the ellipse approximation. Note that when the major and minor axes are equal, it will result in a perfect quarter circle, as suggested by Glassman [16]. The volume of the meridional channel, based on the ellipse approximations, is 4.8% larger than that of the original contour. This difference is small enough to assume that the approximation is sufficiently accurate for predicting turbine weight using the surrogate model.

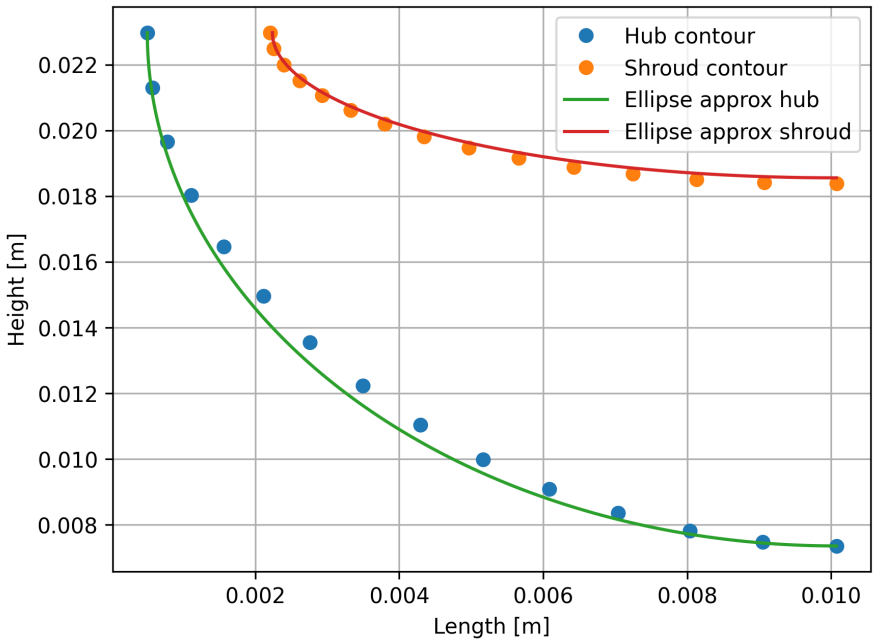


Figure 4.2: Meridional channel contour using the ellipse approximation method from Glassman [16]

When the shapes of all the parts of the geometrical domain are determined, their volume can be calculated. The volumes of the backplate (part 1A and 1B), the stator blades (part 2) and the backface (part 3A) can be calculated using the simplified disk integration, Equation 4.1.The volumes of the disk part of the blisk (part 3B) and the shroud casing are determined with Equation 4.2. By taking a small increment in the axial direction (dx), the risk of over- or underestimating the volume is reduced. It was chosen divide the shapes up into 100 sections. Finding the most optimal contour of the rotor blades requires many extra steps in the calculation process. Similarly as for the stator blades, the volume of the rotor blades is based on simple assumptions. The ORCHID impeller is used as the base case. The ORCHID is the Organic Rankine Cycle Hybrid Integrated Device located at the aerospace faculty [39]. The volume of one ORCHID impeller blade is about 1% of the total impeller weight. Assuming that this is the case for most radial-inflow turbines, the total blade volume of the impeller blisk can be determined by

$$V_{blades} = \frac{\frac{Z_{rot}}{100}}{1 - \frac{Z_{rot}}{100}} \cdot (V_{disk} - V_{shaft}), \tag{4.7}$$

where Z_{rot} is the number of rotor blades, V_{disk} the volume of the disk part of the blisk (including the backface), and V_{shaft} is the shaft volume. The impeller design includes a 4 mm bore to fit the part on the rotor shaft, thus the corresponding volume should be subtracted from the impeller volume.

$$V_{shaft} = \pi \cdot 0.004^2 \cdot (t_{bf} + L_{ax}) \tag{4.8}$$

The total volume of the radial-inflow impeller blisk can be found by applying:

$$V_{blisk} = V_{disk} - V_{shaft} + V_{blades}. \tag{4.9}$$

Additionally, the ORCHID has a locking ring to secure the impeller blisk to the shaft. It is 2 mm thick and 2.5 mm long, the volume of this locking ring can be estimated to be

$$V_{locking\ ring} = \pi \cdot (0.021^2 - 0.019^2) \cdot 0.0025 = 6.28 \cdot 10^{-7}\ m^3 = 628\ mm^3. \tag{4.10}$$

The weight of each component can be found by multiplying the volume by the density of stainless steel ($\rho = 7850$ kg/m^3). Since the turbine will consist only of stainless steel, the total weight of the turbine can be determined by taking the total volume of the turbine and multiplying it by the density.

$$W_{turbine} = \rho \cdot \sum V \tag{4.11}$$

The turbine weight distribution is presented in Figure 4.3, with a total assembly weight of 0.371 kg. The backplate and shroud casing are dominating the total weight, accounting for 63.6% and 23.1%, respectively, while the impeller makes up only 10.2% of the total weight. Since the thickness of the thickest part of the backplate and the thickness of the shroud casing are fixed, their influence on the total weight may vary. It is expected that for significantly larger turbines, this ratio will decrease and the impeller will constitute a larger portion of the total weight.

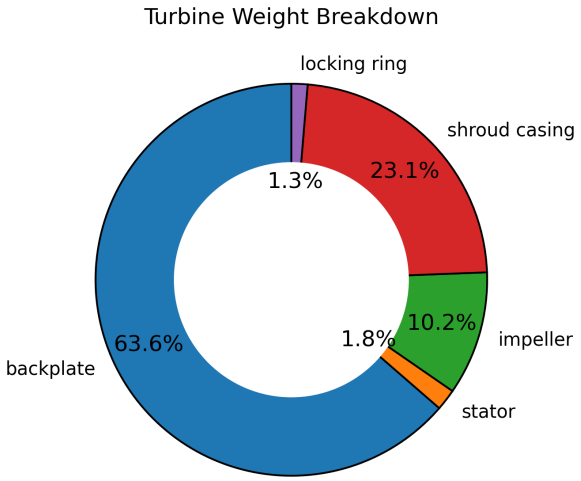


Figure 4.3: Weight distribution of a RIT based on the ORCHID settings using the simplified turbine architecture

4.2. Parametric Study

The TurboSim model requires many input variables, namely fluid, reduced temperature and pressure, mass flow rate, volumetric flow ratio (*VR*), stator inlet angle (α_0), rotor outlet angle (α_3) and 17 geometric variables. The number of design variables to include in the surrogate model was chosen using a sensitivity study, which was designed with the purpose of identifying the most relevant variables. Conducting this sensitivity analysis provides two main advantages. First, including all design variables in the training dataset for the surrogate model would result in a substantial increase in training time. Second, the objective of this thesis is to assess the feasibility of developing a highly accurate surrogate model. Thus, it is more effective to begin with a smaller set of design variables. If this initial approach satisfies the predefined accuracy criteria, the model can then be expanded to incorporate additional variables.

The investigated design variables are tabulated in Table 4.1.

Table 4.1: Description of design variables

Symbol	Description
α_0 and α_3	Flow angles at stator inlet (location 0) and exducer (location 3)
$(g/h)_{bf}, (g/h)_{le}$ and $(g/h)_{te}$	Gap ratios located at the backface, leading edge and trailing edge
$H/\Delta R$	Aspect ratio
$Lax/\Delta R$	Axial-to-radial length ratio
$\phi_{2,is}$	Isentropic flow coefficient at inducer (location 2)
ψ_{is}	Isentropic head coefficient
R_1/R_0, R_2/R_1, R_3/R_2	Radius ratios
R_h/R_t	Hub-to-tip radius ratio
$(t/s)_{st}$ and $(t/s)_{rot}$	Stator and rotor thickness ratios
V_m ratio	Meridional velocity ratio

The values of the design parameters are chosen according to radial turbine design best practices, with the mass flow rate set to 3 kg/s and the volumetric flow ratio to 20. These values are listed inTable 4.2 and 4.3.

The parametric study was performed by only varying one design variable in 10 steps between the minimum and maximum value while keeping the other design variables constant at their average value. The sensitivity of a parameter is determined by calculating the difference between the minimum and maximum values in the dataset, $\Delta\eta$ for efficiency and ΔW for weight.

To enable a meaningful comparison between the efficiency and weight results, the percentage increase is taken as the relative change, calculated by dividing the difference between the maximum and minimum values by the minimum value, and multiplying by 100.

It is assumed that the complete turbine is made out of the same material, stainless steel, and its weight can be estimated by using Equation 4.11.

Table 4.2: Fixed parameters in parametric study

Parameter	Value	Unit
Working fluid	MM	-
P_r	0.937	-
T_r	1.105	-
VR	40.83	-
$\dot{m}$	0.132	kg/s

Table 4.3: Varying design variables for parametric study

Design variable	Min value	Max value	Design variable	Min value	Max value
α_0	5°	45°	ψ_{is}	0.6	1.3
α_3	-10°	20°	R_1/R_0	0.65	0.9
$(g/h)_{bf}$	0.01	0.1	R_2/R_1	0.8	0.975
$(g/h)_{le}$	0.01	0.2	R_3/R_2	0.4	0.6
$(g/h)_{te}$	0.01	0.1	R_h/R_t	0.2	0.6
$H/\Delta R$	0.2	0.4	$(t/s)_{rot}$	0.03	0.1
$L_{ax}/\Delta R$	0.7	1.2	$(t/s)_{st}$	0.01	0.05
$\phi_{2,is}$	0.3	0.5	V_m ratio	1.2	2.0

The results of the parametric study, presented in Table 4.4, reveal notable differences in the influence of design variables on efficiency and weight. For efficiency, the six most influential parameters (indicated in red), ranked from highest to lowest, are R_3/R_2, R_h/R_t, $L_{ax}/\Delta R$, $\phi_{2,is}$, ψ_{is} and $(g/h)_{le}$. Conversely, the parameters with the greatest impact on weight are R_3/R_2, R_h/R_t, $\phi_{2,is}$, Vm ratio, R_1/R_0 and ψ_{is}.

It was observed that most design variables exhibit a stronger influence on weight compared to efficiency. The design variables are geometrical parameters, which will therefore have a stronger affect the weight. This difference is particularly evident in the magnitude of the relative changes calculated during the study. The maximum percentage increase found for efficiency was 33.47%, while for weight it equals 414.38%. Such findings emphasize the importance of accounting for these variations when training a surrogate model, as the relative sensitivity of parameters directly affects the accuracy of predictions for efficiency and weight.

Table 4.4: Results parametric study for efficiency and weight. The six most influential design variables are indicated in red

Parameter	$\eta_{\mathbf{min}}$ [%]	$\eta_{\mathbf{max}}$ [%]	$\Delta\eta$ [%]	**Percentage increase** η [%]	$\mathbf{W_{min}}$ [kg]	$\mathbf{W_{max}}$ [kg]	$\Delta\mathbf{W}$ [kg]	**Percentage increase W** [%]
α_0	81.34	81.36	0.02	0.02	3.48	3.49	0.01	0.08
α_3	78.73	82.86	4.13	5.24	3.16	3.96	0.80	25.23
$(g/h)_{bf}$	81.32	81.40	0.08	0.09	3.47	3.51	0.04	1.25
$(g/h)_{le}$	78.72	83.73	5.01	6.37	3.28	3.70	0.42	12.80
$(g/h)_{te}$	81.09	81.62	0.53	0.65	3.41	3.49	0.08	2.42
$H/\Delta R$	81.34	81.34	0.00	0.00	3.49	3.49	0.00	0.00
$L_{ax}/\Delta R$	76.70	85.46	8.76	11.43	2.99	4.17	1.18	39.55
$\phi_{2,is}$	75.81	83.48	7.68	10.13	2.52	7.34	4.83	91.77
ψ_{is}	76.13	83.38	7.26	9.53	2.77	4.10	1.34	48.27
R_1/R_0	81.29	81.37	0.08	0.10	2.73	4.72	1.99	73.03
R_2/R_1	81.27	81.46	0.19	0.24	3.00	4.15	1.15	38.33
R_3/R_2	66.75	89.09	22.34	33.47	1.87	9.62	7.75	414.38
R_h/R_t	73.76	86.93	13.17	17.86	1.77	8.37	6.59	371.77
$t_{s_{st}}$	79.79	82.47	2.68	3.36	3.41	3.49	0.07	2.19
$t_{s_{rot}}$	80.39	80.39	1.68	2.08	3.41	3.54	0.13	3.74
V_m ratio	78.82	83.24	4.42	5.61	2.77	5.01	2.24	80.68

4.3. Model Development

A surrogate model can be best defined as an approximation model used to simulate the behavior of a more complex and computationally expensive system. The surrogate fits the input and output data to combinations of simple functions to approximate the behavior of the system which it mimics. Figure 4.4 shows the basic concept of surrogate modeling when the system is computationally expensive (slow computation time). The surrogate model, which is trained with data obtained from the system, is able to predict results faster, reducing the computational cost of the system.

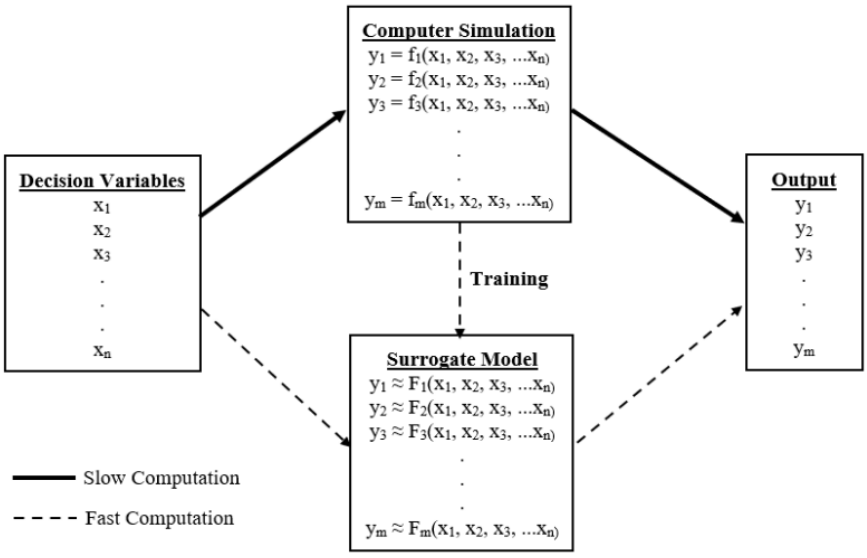


Figure 4.4: Basic concept of surrogate modeling [26]

4.3.1. Surrogate Model Input Parameters

Four parameters, namely the molecular complexity (N), the compressibility factor (Z), the volumetric flow ratio (*VR*), and the mass flow rate ($\dot{m}$) were chosen for the surrogate input layer. Molecular complexity is a constant value for each fluid, allowing the model to distinguish between different working fluids based on this number and account for their thermodynamic behavior. The compressibility factor captures deviations from ideal gas behavior, ensuring that real fluid effects are considered. The volumetric flow ratio is a key parameter in turbine performance, reflecting expansion characteristics and efficiency trends. Finally, the mass flow rate directly influences power output and operational characteristics, making it essential for accurate predictions.

Working fluid selection

The surrogate model will be trained with data from different working fluids, making it a flexible system. Krempus et al. [30], listed 26 working fluid candidates for air-cooled organic Rankine cycle (ORC) bottoming power plants of gas turbines, containing refrigerants, hydrocarbons, siloxanes and perfluorochemical (PFC) fluids. Four working fluids are selected covering all except PFCs categorizations: refrigerant R134a, hydrocarbons cyclopentane and toluene, and siloxane MM. Ethanol is added to investigate the effect of alcohols. Lastly, butane completes the list. It was selected because it has a much lower critical temperature than cyclopentane, even though the molecular weight is similar.

The chemical properties of the investigated working fluids are tabulated in Table 4.5.

Table 4.5: Chemical properties of the investigated working fluids [5, 33]

Working fluid	Chemical composition	Molecular weight [g/mol]	Critical temperature [K]	Critical pressure [bar]	Maximum temperature [K]
Ethanol	C_2H_6O	46.07	514.71	62.68	650
Butane	C_4H_{10}	58.12	425.13	37.96	575
Cyclopentane	C_5H_{10}	70.13	511.72	45.83	550
Toluene	C_7H_8	92.14	591.75	41.26	700
R134a	$C_2H_2F_4$	102.03	374.21	40.59	455
MM	$C_6H_{18}OSi_2$	162.38	518.70	19.31	580

In an ideal scenario, the surrogate model is trained with all fluid data simultaneously, enabling the use of a single model across different working fluids. However, critical properties such as the critical temperature and pressure, as well as the maximum temperature of a fluid, can vary significantly depending on the working fluid. Therefore, it is necessary to investigate whether it is feasible to develop a model that is sufficiently flexible to accurately predict turbine efficiency and weight across these varying fluids. If it is not possible, every working fluid will have its own unique surrogate model, which in return removes the molecular complexity input parameter from the training dataset.

Compressibility factor

The compressibility factor (Z) shows the deviations from ideal gas behavior, since it equals one when a fluid is in the ideal gas state. A fluid is in a supercritical state when both the temperature and pressure are above the vapor-liquid critical value, T_{crit} and P_{crit}, respectively [18]. This study will cover working fluids operating in the dense vapor region for which the reduced pressure is below 1, as well as supercritical fluids for which the reduced pressure is above 1. For each fluid, the values of the reduced total turbine inlet temperature (T_{t0_r}), of the minimum cycle temperature (T_{3_r}), i.e., the condensation temperature, and of the reduced total turbine inlet pressure were computed as follows:

$$T_r = \frac{T}{T_{crit}} \tag{4.12}$$

$$P_r = \frac{P}{P_{crit}} \tag{4.13}$$

where T and P are the temperature and pressure. T_{crit} and P_{crit} are the critical temperature and pressure. The properties were estimated using the REFPROP thermodynamic libraries [33]. They are also stated in Table 4.5. The turbine inlet and outlet temperatures, as well as the critical and saturation temperatures are visualized on a T-s diagram in Figure 4.5.

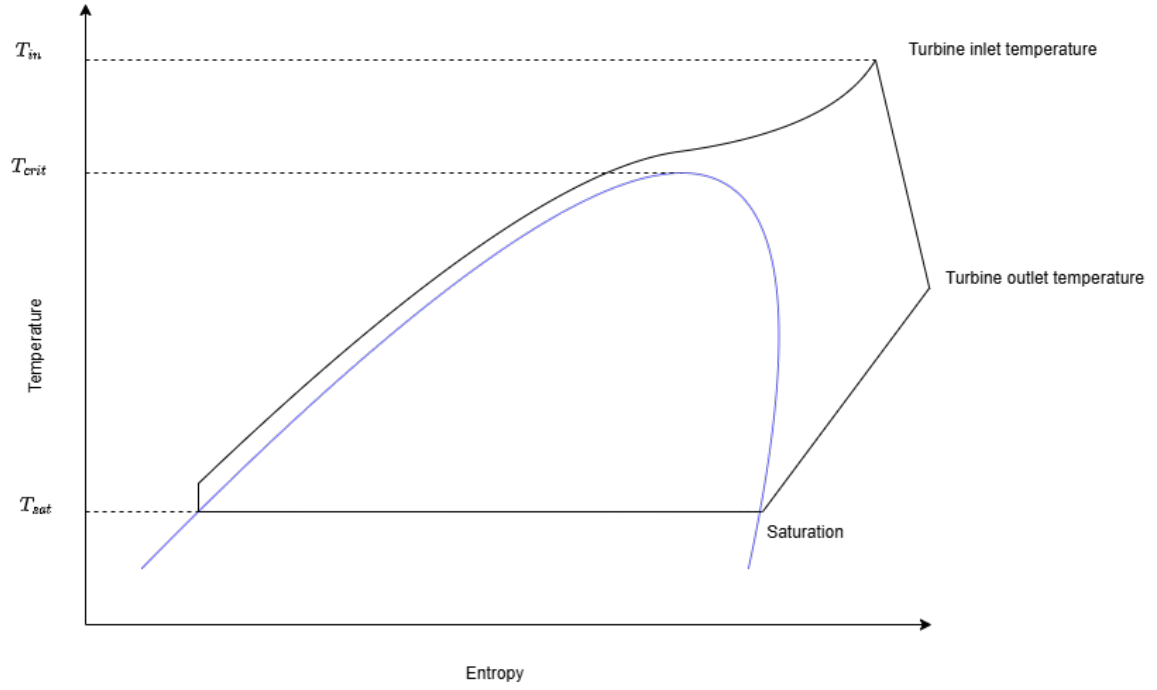


Figure 4.5: T-s diagram

The compressibility factor corresponding to the chosen total turbine inlet pressure and temperature ranges was computed by discretizing the interval of each quantity with 10 values, and combining them

to obtain 100 values of the compressibility factor. This large number of points ensures a uniform distribution of Z values across the specified range.

The process of determining those ranges is explained below. Table 4.6 displays the compressibility factor ranges (lower bound (LB) and upper bound (UB)) for each investigated working fluid.

Table 4.6: Compressibility factor ranges for different working fluids

Working fluid	**Z** [-]	
	LB	UB
Butane	0.1	1.0
Cyclopentane	0.1	0.8
Ethanol	0.1	1.0
MM	0.1	0.9
R134a	0.1	0.9
Toluene	0.1	0.9

Reduced temperature

The temperature ranges chosen for this work are based on the previous work of Krempus et al. [28], where the condensation temperature bounds were set to 323 and 423 K, while the total turbine inlet temperature ranged between $T_{t0_{UB}} \in (T_{max} - 100\ [K], T_{max})$. It is important that the condensation temperature remains below the supercritical threshold.

For R134a, the upper bound of the condensation temperature lies within the supercritical range, while for butane, the reduced condensation temperature is 0.995. As a result, a temperature range that accommodates all the investigated working fluids must be identified. After testing various temperature ranges, it was decided to set the upper bound of the reduced condensation temperature to $T_{cond_r} = 0.9$, ensuring that the temperature remains below the supercritical threshold.

The total turbine inlet temperature (T_{t0}) range of the investigated working fluids, cannot overlap with the condensation temperature (T_{cond}). When this is the case, there is no thermodynamic cycle possible. To be able to investigate the complete range of the compressibility factor ($0 \leq Z \leq 1$), the maximum cycle inlet temperature range needs to cover both sub- and supercritical temperatures, while still complying with $T_{t0_r} > T_{cond_r}$. It was found that when using a 100 K temperature range for T_{t0} resulted in supercritical lower bounds for butane, ethanol and toluene. Hence, these T_{t0_r} values were lowered to $T_{t0_r} = 0.95$. Ensuring the inlet temperature range covers both sub- and supercritical temperatures for every working fluid. These temperature ranges are tabulated in Table 4.7.

Table 4.7: The reduced temperatures used in TurboSim and its corresponding temperatures expressed in Kelvin. The temperatures deviating from the initial temperature range are highlighted in blue

Working fluid	$\mathbf{T_{cond_r}}$ [-]		$\mathbf{T_{t0_r}}$ [-]		$\mathbf{T_{cond}}$ [K]		$\mathbf{T_{t0}}$ [K]	
	LB	UB	LB	UB	LB	UB	LB	UB
Butane	0.76	0.90	0.95	1.35	323	382	404	575
Cyclopentane	0.63	0.83	0.88	1.08	323	423	450	550
Ethanol	0.63	0.82	0.95	1.26	323	423	489	650
MM	0.62	0.82	0.93	1.12	323	423	480	580
R134a	0.86	0.90	0.95	1.22	323	336	355	455
Toluene	0.55	0.72	0.95	1.18	323	423	562	700

Reduced Pressure

In TurboSim, the reduced pressure input parameter corresponds to the reduced condensation pressure. While Krempus et al. [28] considered only supercritical pressures, the surrogate model incorporates both sub- and supercritical pressures. As a result, the reduced condensation pressure range for this model is set between 0.75 and 1.25.

Molecular complexity

The molecular complexity N of the fluid was determined according to the following definition [8]:

$$N = \frac{2C_{v,id}(T_c)}{R}, \tag{4.14}$$

where $C_{v,id}(T_c)$ is the ideal gas specific heat at constant volume evaluated at the critical temperature T_c and R is the universal gas constant. The fluid with the lowest molecular complexity is ethanol (alcohol), the refrigerant R134a follows closely, then it is the hydrocarbons and lastly, siloxane MM is the most complex fluid. These results are shown in Table 4.8.

Table 4.8: Molecular complexity of investigated working fluids

Working fluid	**N**
Ethanol	23.52
R134a	23.64
Butane	31.52
Cyclopentane	37.19
Toluene	46.76
MM	80.36

Volumetric flow ratio

The volumetric flow ratio (*VR*) can be determined by using Equation 4.15

$$VR = \frac{P_{t_0}}{P_{liq,min}}, \tag{4.15}$$

where P_{t_0} is the total pressure at the inlet, which is calculated by using Equation 4.16 and $P_{liq,min}$ is the vapor saturation pressure.

$$P_{t_0} = P_r \cdot P_{crit} \tag{4.16}$$

The *VR* ranges can be found in Table 4.9. There is a large difference between the working fluids, indicating that creating an accurate surrogate model that is trained with data from all working fluids at once is not possible. The solution is to create a separate model for every working fluid.

Table 4.9: VR ranges for the investigated working fluids

Working fluid	**VR** [-]	
	LB	UB
Butane	1.56	9.61
Cyclopentane	2.94	55.43
Ethanol	4.80	268.36
MM	4.15	138.34
R134a	1.66	3.86
Toluene	11.28	422.43

Mass flow rate range selection

The effect of mass flow rate on net total-to-total efficiency ($\eta_{tt_{net}}$), size parameter (*VH*) and power (P_w) was investigated by running TurboSim while only specifying the necessary input parameters and using predefined values for the other parameters. MM was selected as the working fluid, with the reduced temperature and pressure were set to 1.105 and 0.937, respectively, similarly to the parametric study. The mass flow rate varied from 0.5 kg/s to 5 kg/s, while *VR* was set to 3.0, as it lies within the *VR* ranges of all fluids (see Table 4.9). Note that TurboSim cannot find a solution for every investigated design because some combinations violate thermodynamic property limits.

The mass flow rate had no significant impact on efficiency, as is shown in Figure 4.6. However, an increase in *VH* and P_w was observed in Figure 4.7 and 4.8, respectively. Since the size parameter is indirectly related to turbine weight, a lower *VH* indicates a more compact design, while a larger *VH* corresponds to a heavier turbine. Therefore, a mass flow rate range of 0.5–5.0 kg/s was selected as it provides sufficient variation to analyze its influence on turbine design.

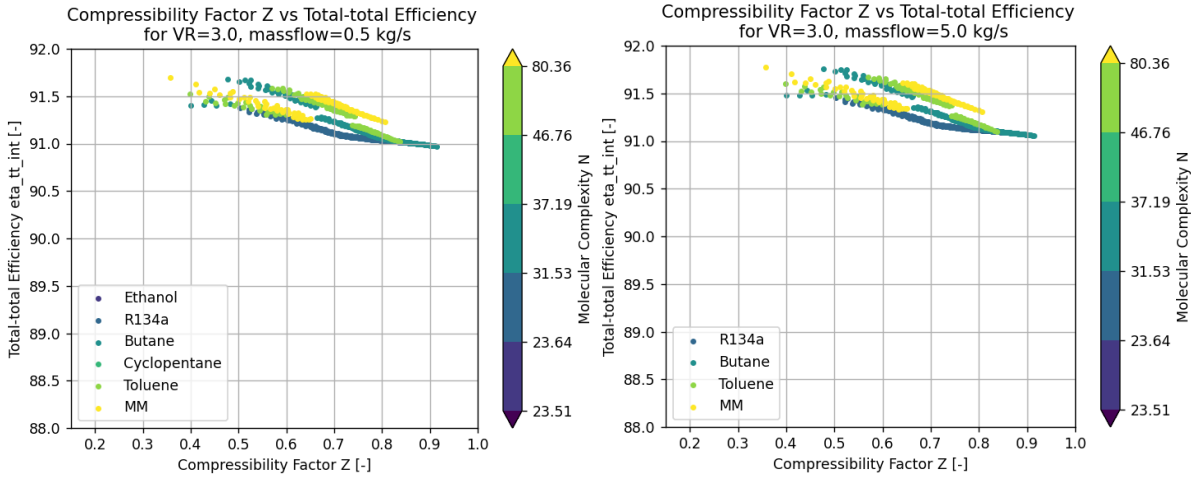


Figure 4.6: Effect of mass flow rate on efficiency ($\eta_{tt_{net}}$)

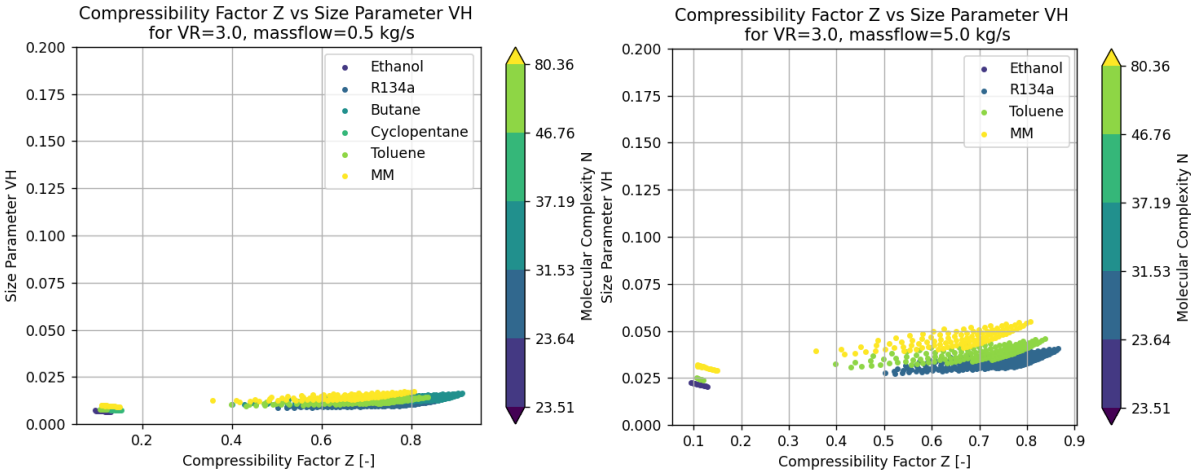


Figure 4.7: Effect of mass flow rate on size parameter (*VH*)

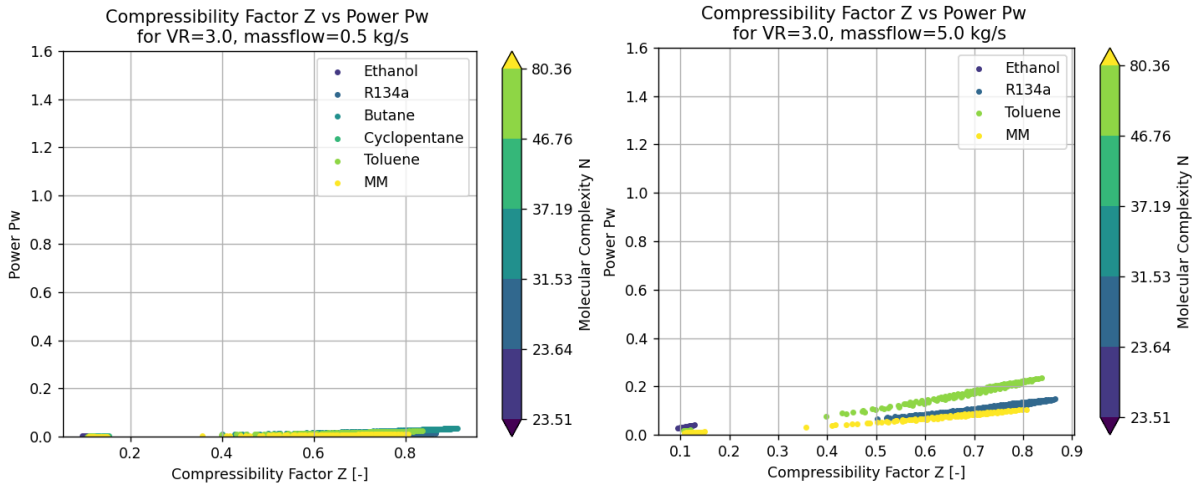


Figure 4.8: Effect of mass flow rate on power (P_w)

4.3.2. Design Variables

The number of design variables included in the training dataset for the surrogate model, will have a large influence on the required time allocated to the data generation. The data generation is performed by using TurboSim, which can take anywhere between a couple of seconds to hundreds of seconds to generate a result. This comes from how TurboSim is created. It has iteration loops, with stopping criteria to prevent it ending up in an endless loop. However, not every tested combination of input parameters has a result, making it difficult to have an accurate time estimation for the data generation. The estimated time required to collect the training data for the surrogate model is calculated below, assuming an average computation time of 2 seconds per result.

Six surrogate models will be created, one per working fluid. To ensure a uniform coverage of the compressibility factor, 100 points are investigated. The next step is to determine how many values per variable are needed to get a good prediction. Giuffre et al. [15] generated a data-driven model for high-speed centrifugal compressors. The dataset consisted of $240\,000$ unique compressor stage designs, obtained by varying 10 design parameters and for 8 different fluids. Thus, this yields $240\,000/8 = 30\,000$ compressor configurations for each fluid. To compute the discretization of each design parameter one can use the following formula:

$$\#steps = \sqrt[10]{30000} = 2.8 \approx 3 \text{ steps per parameter.}$$

Using this result for the present work, which includes 6 working fluids, 100 values of Z, 3 mass flow rates, 3 volumetric flow ratios, and a design variables, yields:

$$\#\text{investigated configurations} = 6 \cdot 100 \cdot 3^2 \cdot 3^a$$

As mentioned above, one configuration is assumed to take 2 seconds to generate. Table 4.10 lists the number of unique turbine configurations based on how many varying design variables are included in the data generation and an estimate of the generation time on the Delft Blue Supercomputer using 36 processing cores. One can see that the minimum generation time increases exponentially with the number of design variables, as well as the complexity of the surrogate model. To evaluate the feasibility and accuracy of the surrogate model, only a limited number of DVs are included. This approach maintains flexibility in the turbine design space while avoiding unnecessary complexity.

Table 4.10: Details on how large the data generation dataset will be and how long the generation will take on the Delft Blue Supercomputer

# DVs	# configurations	Minimum generation time on 36 cores	
		[h]	[days]
4	437 400	6.75	0.28
5	1 312 200	20.25	0.84
6	3 936 600	60.75	2.53
7	11 809 800	182.25	7.59

Using the results from the parametric study, it was decided to include the six highest scoring design variables on percentage increase, which makes up about one-third of number of design variables. The most influential DVs for efficiency are, from highest to lowest influence, R_3/R_2 (33.47%), R_h/R_t, $L_{ax}/\Delta R$, $\phi_{2,is}$, ψ_{is} and $g_{h_{le}}$ (6.37%). However, the six most influential design parameters for weight are R_3/R_2 (414.38%), R_h/R_t, $\phi_{2,is}$, Vm_{ratio}, R_1/R_0 and ψ_{is} (48.27%). Considering that the surrogate will be trained for efficiency and weight separately, as well as simultaneously, the selected parameters will need to be the same for all cases. Therefore, the parameters with the highest influence on weight weight are selected. The settings for the data generation are presented in Table 4.11 and 4.12.

Table 4.11: Design variable ranges used to generate training data in TurboSim

Design variable	Value	
	Minimum	Maximum
Head coefficient ψ_{is}	0.8	1.3
Flow coefficient $\phi_{2,is}$	0.25	0.5
Radius ratio nozzle R_1/R_0	0.65	0.9
Radius ratio impeller R_3/R_2	0.4	0.6
Hub-to-tip radius ratio R_h/R_t	0.2	0.6
Meridional velocity ratio V_m ratio	1.2	1.6

Table 4.12: Fixed design variables to generate training data in TurboSim

Design variable	Value
α_0	10°
α_3	5°
$(g/h)_{le}$	0.1
$(g/h)_{te}$	0.05
$H/\Delta R$	0.3
$L_{ax}/\Delta R$	1.0
R_2/R_1	0.9
$(t/s)_{rot}$	0.05

4.3.3. Data Generation and Post Process

The next step is to generate the training data using TurboSim. As mentioned above, it was decided to investigate 3 values per design variable, mass flow rate and volumetric ratio. In total, $3\,936\,600$ cases will be tested, resulting in $656\,100$ cases for every working fluid. The overall data generational time can be significantly reduced when the test cases are divided into the 9 unique combinations of mass flow rate and volumetric flow rate. These were ran in parallel on the Delft Blue Supercomputer.

The first step of the post processing process is to combine the 9 generated datasets into one. The next step is to decide how the data will be used, there are two ways: 1. keep it as is (complete dataset), 2. apply constraints to filter out unfeasible cases (reduced dataset).

The reduced dataset was created by specifying 13 constraints (Equation 4.17 - 4.29). The numbers written in square brackets indicate the location along the cross-section. For each location, the distance between the hub and tip is divided into 5 sections. Location [x,0] indicates the hub, location [x,2] is at the mid point and location [x,-1] is located at the tip.

The first three constraints are related to the blade angles at the stator outlet α_1 (location 1), impeller inlet or inducer β_2 (location 2) and impeller outlet or exducer β_3 (location 3). To limit the risk of flow separation, the blade angles at the stator outlet and impeller inlet need to be smaller than $85°$ and $45°$, respectively. The exducer blade angle is required to be larger than $-65°$. The difference between the hub and tip exducer blade angles ($\Delta\beta_3$) cannot exceed $30°$. To ensure that the flow at the exducer is at most transonic, the relative Mach number was limited to 1.2. The tip velocity U_2 at the impeller inlet cannot exceed 650 m/s. A higher tip velocity would expose the rotor blades to higher stresses, which could result in material fatigue and failure of the blades. The number of blades is limited to 20 and

30 for rotor and stator blades respectively, to limit the amount of added weight on the rotor.The blade thickness was chosen to be above 0.5 mm, to ensure manufacturability of the blades. A minimal tip clearance of 0.05 mm is chosen to allow room for vibrations that occur during operation. The minimal blade height should exceed 2 mm and the blade radius should be at least 4 mm. Lastly, the difference between the radius at the stator and inducer taken at the mid point should be at least 1 mm.

$$\frac{\alpha[1,2]-85}{85}<0 \tag{4.17}$$

$$\frac{\beta[2,2]-45}{45}<0 \tag{4.18}$$

$$-\left(\frac{\beta[3,2]+65}{65}\right)<0 \tag{4.19}$$

$$\frac{M_{rel[3,-1]-1.2}}{1.2}<0 \tag{4.20}$$

$$\frac{U[2,-1]-650}{650}<0 \tag{4.21}$$

$$\frac{Z_{rot}-20}{20}<0 \tag{4.22}$$

$$\frac{Z_{st}-30}{30}<0 \tag{4.23}$$

$$\frac{|\beta[3,0]-\beta[3,-1]|-30}{30}<0 \tag{4.24}$$

$$-\left(\frac{min(t)-0.0005}{0.0005}\right)<0 \tag{4.25}$$

$$-\left(\frac{min(g)-0.00005}{0.00005}\right)<0 \tag{4.26}$$

$$-\left(\frac{min(H)-0.002}{0.002}\right)<0 \tag{4.27}$$

$$-\left(\frac{min(R)-0.004}{0.004}\right)<0 \tag{4.28}$$

$$\frac{0.001-(R[1,2]-R[2,2])}{0.001}<0 \tag{4.29}$$

The dataset was additionally filtered to remove design points for which the loss models converged to non-physical values, causing unlikely high values of the turbine efficiency (net total-to-total), which in some cases resulted in a 100% efficiency. It was observed that applying the constraints, except for 5 cases of working fluid ethanol, removed these non-physical cases as well.

The next step involves splitting the dataset into training data and test data. This was done to assess whether the equations generated by PySR produce consistent results when applied to data that was not used in training. A 90/10% split of the data was selected over an 80/20% split due to the large size of the dataset. This ensures that the model can be trained with the highest number of datapoints, while keeping a large enough dataset size to perform a meaningful validation. A similar approach was taken by Giuffré et al. [15], who allocated 6% of the dataset for development and 6% for testing, leaving 88% of the dataset for training the surrogate model. This is comparable to the selected split in this study. The validation step of the surrogate model is computing the R-squared value for the test data also called 'new data' and compare it to the R^2 value of dataset of the same size randomly selected from the training dataset ('seen data'). More information on the R^2 method can be found in Appendix A.

4.3.4. PySR

From the literature study performed on machine learning algorithms, see section 2.2, it was determined to use symbolic regression. The advantage of using symbolic regression over a neural network is that it is not a black-box model, making the resulting expressions more interpretable. Unlike neural networks, which require extensive training data and can struggle with extrapolation beyond the training dataset, symbolic regression generates explicit mathematic expressions that provide insight into the underlying relationships between variables. These expressions can easily be analyzed, and implemented in various applications without the need for retraining. Additionally, symbolic regression can uncover fundamental physical relationships within the data, making it especially valuable in engineering and scientific applications.

The best suitable off-the-shelf python package is PySR. The changed PySR settings are listed in Table 4.14. A more extensive list, including the default settings, can be found on the PySR website[3]. The optimizer algorithm used is the Broyden-Fletcher-Goldfarb-Shanno or BFGS algorithm, which is a widely used second-order optimization algorithm. PySR uses a loss function created in Julia that

[3] https://astroautomata.com/PySR/api/

measures the squared L_2 distance between predicted and target values,

$$L_2 \text{ Distance Loss} = \sum_{i=1}^{N} (y_{true,i} - y_{pred,i})^2. \tag{4.30}$$

It is similar to the mean-squared-error, as can be seen in

$$MSE = \frac{1}{N} \sum_{i=1}^{N} (y_{true,i} - y_{pred,i})^2. \tag{4.31}$$

The default model selection criterion is set to 'best,' which chooses the expression with the highest score among those that achieve a loss no more than 1.5 times greater than the loss of the most accurate model. This setting is maintained to avoid overcomplicating the model, ensuring that the selected expression remains both effective and efficient. The number of cycles per iteration (380) is the number of total mutations to run, per 10 samples of the population, per iteration. This setting was untouched.

When test cases were run, it became clear that the standard settings were not ideal for the large training dataset. The training was very slow (>14 h) because it took the whole dataset into account at once. A solution was to turn 'batching' on. Different testing cases were performed and three cases are tabulated in Table 4.13. It can be seen that the computational time increases with increasing the batch size. However, the Mean Squared Error (MSE) of the models drastically reduced when increasing the batch size. The MSE is used by PySR to indicate the accuracy of the model, more information on MSE can be found in Appendix B. A batch size of 2000 came out to be the best time-accuracy combination, since it required only half of the training time required for a batch size of 4000, while achieving a better MSE value.

Table 4.13: Testing different batch sizes on accuracy and computational time

Batch size	**Computational time** [min]	**MSE**
50	7.1	11.3
2000	20	7.5
4000	49	8.7

Unary operators such as *sin*, *cos*, *tan*, *ln* and *exp* were included to enable more complex expressions. Additionally, the power operator ('^') was added to the set of binary operators, expanding the search space for potential solutions. The maxsize was increased from 30 to 40 and 60, so that it has more room to explore. The surrogate model is trained on 9 parameters (compressibility factor, mass flow rate, volumetric flow ratio and the 6 selected DVs), which would result in a minimal complexity of 17 (9 parameters and 8 operators) when only the simple binary operators are used and if all parameters are included in the expression. The number of iterations was lowered to 40, to reduce the training time, while the population size was slightly increased from 27 to 33, to allow for more individuals per population. The creators of PySR suggested running the training of the surrogate model in parallel on multiple processing cores and set 'populations' = $3 \cdot$ # cores. The surrogate was run on the Delft Blue Supercomputer as well, and 20 cores were used. Thus, each training would have 60 populations. Due to the fact that unary operations were used, constraints need to be specified. The three types of constraints stated below were tested separately.

1. Only self nesting is allowed up to a depth of 1, meaning that $\sin(\sin(x))$ is allowed
2. No nesting of any sorts is allowed
3. No constraints are specified

Each time PySR is run, the model is initialized from scratch, meaning that different results may be obtained even when using the same settings. To improve the consistency of the results, it is advisable to keep the number of operators minimal and avoid redundant ones, such as using both '-' and 'neg', or '^' and 'pow'. However, given that the optimal operator selection is not always known in advance, all

combinations of the operators specified in Table 4.14 are tested, resulting 188 unique testing settings. Additionally, the training code is structured so that every session involves splitting the dataset into training and verification sets, ensuring that each training run uses a unique subset. While this can introduce slight variations in the results, the large size of the dataset mitigates any significant impact on the overall outcomes.

Table 4.14: Changed PySR settings for training surrogate model [3]

Setting	Default	Used
Binary operators	+, -, *, /	+, -, *, /, ^
Unary operators	None	sin, cos, tan, ln, exp, abs[4]
Maxsize	30	40 & 60
Niterations	100	40
Populations	31	60
Population size	27	33
constraints	None	Specified
nested constraints	None	Specified
batching	False	True
batch size	50	2000

[4] Only used for training weight separately using the complete dataset

5

Results and Discussion

This chapter will discuss the results of the study. Section 5.1 discusses the distribution of the datasets used to train the surrogate models, as well as the effect of applying constraints to these different datasets. The performance of the surrogate models are discussed in section 5.2.

5.1. Data Generation and Post-Process

This section is organized as follows. Firstly, subsection 5.1.1 will look at the distribution of mass flow rate and volumetric flow ratio in the different datasets. Secondly, subsection 5.1.2 looks into the distribution of the six selected design variables in the datasets. Lastly, the distribution of efficiency and weight in the datasets is discussed in subsection 5.1.3.

The number of cases for which TurboSim generated a turbine design is tabulated in Table 5.1. It contains the complete dataset, which are the results obtained from TurboSim for which the loss models converged to physical values; and the reduced dataset, which is obtained by filtering the data based on feasibility constraints, which are discussed in more detail in subsection 4.3.3. It can be seen that the obtained datasets are relatively small compared to the number of cases that were tested. This is a direct result from the stopping criteria in the iterative processes in TurboSim to limit the search time to find a design.

Since the dataset was generated using only three discrete values (minimum, mean, maximum) for each design variable, mass flow rate and volumetric flow ratio, intermediate feasible designs may not have been captured. As a result, applying the 13 constraints reduced the dataset size to only a fraction of its original size (1.7% to 15.4%). The percentage of feasible designs relative to the total number of investigated cases per working fluid is even lower (0.8 to 6.7%). Relaxing the constraints had little effect on the reduced dataset size, suggesting that the limited coverage of feasible regions is the primary issue rather than overly restrictive constraints.

Table 5.1: Dataset size of complete and reduced datasets generated by TurboSim

Working fluid	**Complete dataset**		**Reduced dataset**		
	[cases]	[% investigated designs]	[cases]	[% investigated designs]	[% complete dataset]
Butane	358 524	54.6	29 562	4.5	8.2
Cyclopentane	157 762	24.0	17693	2.7	11.2
Ethanol	172 506	26.3	9686	1.5	5.6
MM	287 224	43.8	44 162	6.7	15.4
R134a	304 196	46.4	5257	0.8	1.7
Toluene	290 645	44.3	29 685	4.5	10.2

5.1.1. Distribution of Mass Flow Rate and Volumetric Flow Ratio in Dataset

Table 5.2 shows the dataset breakdown across all tested combinations of mass flow rate and volumetric ratio. It is useful to analyze the datasets obtained from TurboSim to understand how the surrogate models behave after training. It is expected that a surrogate model trained on an equally distributed dataset will predict different results compared to a model that is trained on a skewed dataset.

It can be seen that the mass flow rate does not affect the number of cases generated by TurboSim (complete dataset). However, the volumetric flow ratio has a large affect on the distribution of working fluid ethanol, which is the fluid with the lowest molecular complexity investigated in this research. The dataset size will halve each time the volumetric ratio is increased. A similar trend is observed in the dataset for toluene, where the dataset size drops significantly from the minimum to the mean *VR* but remains nearly the same between the mean and maximum *VR* values. This indicates that TurboSim requires more iterations than the set termination criterion to converge on designs operating at the mean or maximum *VR*. For further research, increasing iteration limit could allow more designs to be included in the dataset. However, increasing the number of iterations can also increase the computational time.

In the complete dataset generation, mass flow rate does not affect the number of cases that are generated by TurboSim. However, the volumetric flow ratio does have an influence on the number of cases. This suggests that the surrogate model will be trained with datasets that are more sensitive to a change in volumetric flow ratio than to mass flow rate. This can have an effect on the performance of the model, which will be discussed in section 5.2.

The data distribution of the reduced datasets show that both mass flow rate and volumetric flow ratio affect the number of cases in the datasets. It can be seen that when $\dot{m} = 0.5$ kg/s (minimum), it will not result in any feasible designs for butane and R134a. The other fluids will not have any feasible design for the combination of minimum mass flow rate and minimum VR. This shows that most designs operating at the minimum mass flow rate do not comply with the specified constraints, stated in subsection 4.3.3. The surrogate models trained on these datasets will be sensitive to the given mass flow rate as well as the volumetric flow ratio.

Table 5.2: Breakdown of dataset size, split into the 9 unique combinations of mass flow rate and volumetric ratio

Working fluid	**VR** [-]	**Mass flow rate** [kg/s]	**Size dataset** [cases]	
			Complete	Reduced
Ethanol (N=23.515)	4.80 (min)	0.5	34559	0
		3.0	34532	1229
		5.0	34531	3629
	136.585 (mean)	0.5	14282	113
		3.0	14280	1379
		5.0	14284	1755
	268.36 (max)	0.5	8680	84
		3.0	8683	653
		5.0	8675	844
R134a (N=23.638)	1.66 (min)	0.5	33558	0
		3.0	33590	58
		5.0	33566	863
	2.76 (mean)	0.5	34136	0
		3.0	34160	253
		5.0	34159	1430
	3.86 (max)	0.5	33639	0
		3.0	33707	496
		5.0	33681	2157
Butane (N=31.515)	1.56 (min)	0.5	38885	0
		3.0	38942	849
		5.0	38903	2933
	5.585 (mean)	0.5	40521	0
		3.0	40508	3333
		5.0	40539	7051
	9.61 (max)	0.5	40064	0
		3.0	40057	5980
		5.0	40105	9416
Cyclopentane (N=37.188)	2.94 (min)	0.5	18065	0
		3.0	18054	325
		5.0	18029	1196
	29.185 (mean)	0.5	18179	16
		3.0	18195	3425
		5.0	18203	5079
	55.43 (max)	0.5	16332	197
		3.0	16352	3321
		5.0	16353	4134
Toluene (N=46.760)	11.28 (min)	0.5	37131	0
		3.0	37153	4475
		5.0	37145	7865
	216.855 (mean)	0.5	30412	1055
		3.0	30460	4277
		5.0	30484	4752
	422.43 (max)	0.5	29287	815
		3.0	29302	3044
		5.0	29271	3402
MM (N=80.355)	4.15 (min)	0.5	29816	0
		3.0	29833	2477
		5.0	29820	5027
	71.245 (mean)	0.5	32921	1324
		3.0	32930	8271
		5.0	32962	9516
	138.34 (max)	0.5	32936	1861
		3.0	33003	7423
		5.0	33003	8263

Butane is considered to investigate the effect of feasibility constraints on the turbine design space, as per Table 5.3. Tables for the other working fluids considered in this study can be found in Appendix B. When more than 50% of the dataset violates a constraint, it is indicated in red. The constraint equations were reported in subsection 4.3.3.

As the mass flow rate increases, the blade height must increase to accommodate the larger flow area required. Therefore, imposing smaller mass flow rates tends to lead to unfeasible turbine designs because of either too low blade thickness, or too low blade height. The blade height constraint (constraint 9) is violated the most in the cases operating at the minimum mass flow rate of 0.5 kg/s. Figure 5.1 shows the minimum blade height plotted against the compressibility factor *Z* for the complete *VR* range and $\dot{m} = 0.5$ kg/s. The design variables were fixed at their respective mean value. The blade height for the cases operating at the smallest *VR* required the largest blades. The blade height is found by applying Equation 5.1:

$$H = \frac{\dot{m}}{2\pi \cdot v_m \cdot \rho \cdot R}, \tag{5.1}$$

where $\dot{m}$ is the mass flow rate, v_m the meridional velocity, ρ the density and R the radius. The *VR*, determined in Equation 4.15, is a pressure ratio which can be used to calculate the density. The equation of state is given in Equation 5.2:

$$\rho = \frac{P}{R \cdot T}, \tag{5.2}$$

with P the pressure, R the gas constant and T the temperature. A smaller *VR* will result in a lower density, therefore resulting in a larger blade height.

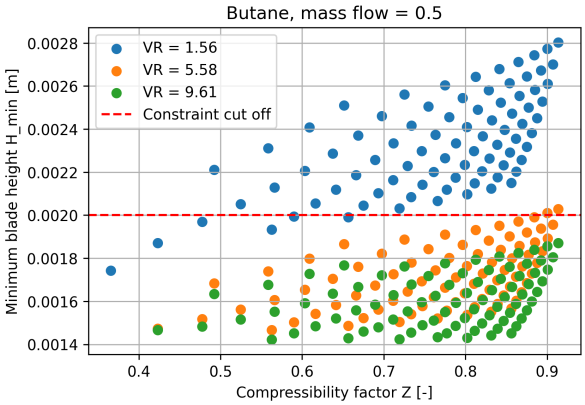


Figure 5.1: Minimum blade height vs compressibility factor for working fluid butane, $\dot{m} = 0.5$ kg/s, DVs are fix at their respective mean value

Two constraints are never violated for any of the investigated working fluids, namely the constraint on the exducer blade angle (constraint 3) at the midpoint between the hub and the tip, and the constraint on the maximum number of stator blades allowed (constraint 6). This is shown in Table 5.3 for working fluid butane and Appendix B for the other working fluids.

The tip blade speed constraint (constraint 5) was necessary for working fluid ethanol. Alcohols, like ethanol, have a low molecular weight compared to hydrocarbons, shown in Table 4.5. The specific heat capacity at constant pressure (c_p) of alcohols is higher than that of hydrocarbons, which results in a higher isentropic work and consequently larger peripheral speeds U_2, following

$$U_2 = \sqrt{\frac{w_{is}}{\psi}}, \tag{5.3}$$

$$w_{is} = c_p \cdot (T_{t0} - T_{t2}). \tag{5.4}$$

One-third of all cases generated by TurboSim required more than 30 stator blades (constraint 7). The mass flow rate and volumetric flow ratio do not affect this percentage. However, the design variable R_1/R_0 does affect the constraint violations, as is shown in Figure 5.2. All designs that operate at the maximum R_1/R_0 value, required more than 30 stator blades. This is the case for all six working fluids.

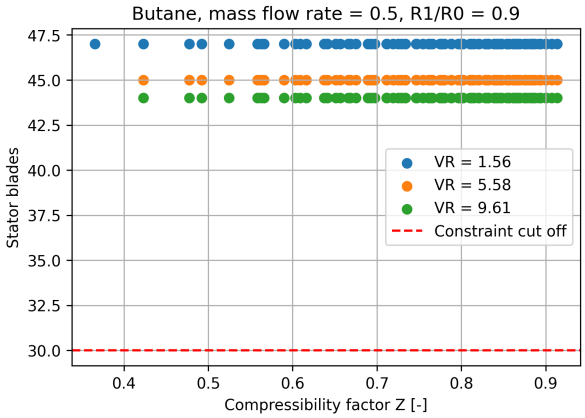


Figure 5.2: Number of stator blades required vs compressibility factor for the complete *VR* range. Working fluid butane, $\dot{m} = 0.5$ kg/s, $R_1/R_0 = 0.9$ (maximum), other DVs are fixed at their respective mean value.

The minimal tip clearance (constraint 10) was chosen equal to 0.05 mm which deemed feasible with typical manufacturing techniques, and assuming that machine vibrations while in operation are comparatively small. Only designs featuring small values of $\dot{m}$ were impacted by this constraint, as the size of the turbine reduces in such cases.

Table 5.3: Constraint violations for working fluid butane expressed in percentage of the number of cases in the complete dataset

Constraint	**VR min**			**VR mid**			**VR max**		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1. α_1	-	-	-	-	-	-	-	-	-
2. β_2	5.13%	5.18%	5.12%	4.90%	4.85%	4.84%	5.00%	5.02%	5.04%
3. β_3	-	-	-	-	-	-	-	-	-
4. M_{rel}	-	-	-	8.10%	8.11%	8.11%	21.38%	21.39%	21.33%
5. $U_{2,tip}$	-	-	-	-	-	-	-	-	-
6. Z_{rot}	-	-	-	-	-	-	-	-	-
7. Z_{st}	33.27%	33.36%	33.24%	33.34%	33.24%	33.32%	33.30%	33.29%	33.32%
8. $\Delta\beta_3$	18.42%	18.44%	18.36%	27.91%	27.91%	27.91%	27.52%	27.47%	27.43%
9. t_{min}	100.00%	97.01%	90.35%	99.99%	89.31%	77.46%	99.85%	79.84%	67.70%
10. g_{min}	-	-	-	0.19%	-	-	0.32%	-	-
11. H_{min}	36.48%	-	-	57.51%	0.85%	0.03%	60.57%	2.06%	-
12. R_{min}	61.70%	19.67%	0.03%	51.03%	10.22%	5.57%	43.25%	7.21%	6.39%
13. $R_1 - R_2$	12.53%	-	-	4.30%	-	-	0.53%	-	-

The most restrictive constraint is the one for the minimum blade thickness (constraint 9). Figure 5.3 shows the blade thickness computed by TurboSim for all six working fluids, sorted in ascending order. A minimum thickness of 0.5 mm was assumed to ensure manufacturability of the blades, as indicated by the dashed horizontal line in the figure. It can be observed that designs operating with fluids of lower molecular complexity (ethanol, R134a, butane and cyclopentane) do not meet this criterion. For the more complex fluids, MM and toluene, approximately 50% of the designs fail to meet the constraint.

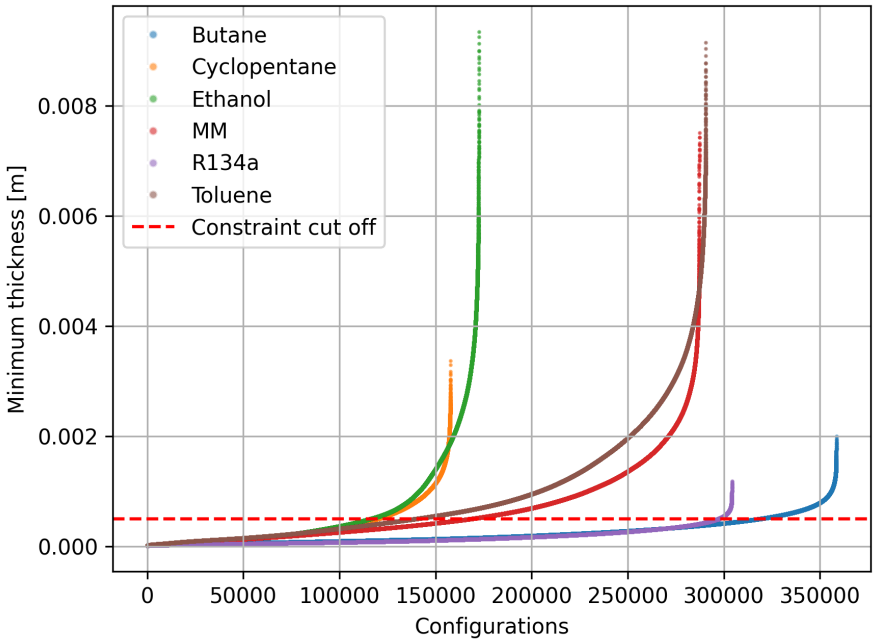


Figure 5.3: Minimum thickness constraint violation for all investigated working fluids

5.1.2. Effect of Design Variables on Dataset

The next step is to examine how the design variables influence the datasets. This is necessary as it provides insight into how the training process might be performed and the potential accuracy of the predictions. Working fluid butane will be discuss in detail, because it has the largest dataset. The dataset distributions of the other working fluids can be found in Appendix C.

Butane

In the complete dataset, as shown in Figure 5.4, the flow coefficient ($\phi_{2,is}$) and radius ratio R_3/R_2 are unevenly distributed. The minimum values of these parameters account for only about 20% of the dataset, while the maximum values make up more than 40%. Since the other parameters are more evenly distributed, this imbalance suggests that the dataset inherently favors certain design regions, potentially impacting future modeling efforts.

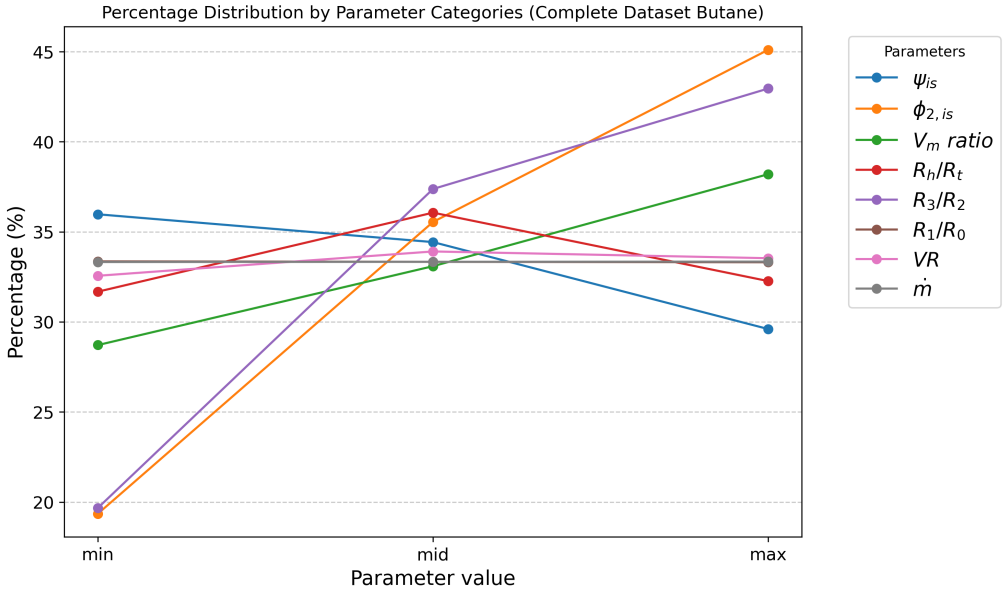


Figure 5.4: Distribution of parameters in complete dataset for Butane

The distribution of the design parameters for the constrained dataset is illustrated in Figure 5.5. The most significant difference compared to the distribution of the entire dataset is observed for the radius ratio of the nozzle (R_1/R_0). As R_1/R_0 increases, the nozzle aspect ratio decreases and so does the nozzle blades thickness and the blade height. The minimum thickness is in most cases lower than the minimum feasible value based on manufacturing considerations, which corroborates with the reduction in the number of feasible designs when considering larger values of R_1/R_0. Similarly, the radius ratio R_3/R_2 excluded certain cases at its maximum value due to constraint violations.

The distribution of the remaining design variables appears to be largely unaffected by the constraints. However, the reduced dataset has a significantly different distribution compared to the complete dataset. As a result, the relative importance given to these portions of the dataset will be different during training. The accuracy of the model can improve or decrease depending on the shape of the dataset. This will be discussed in section 5.2

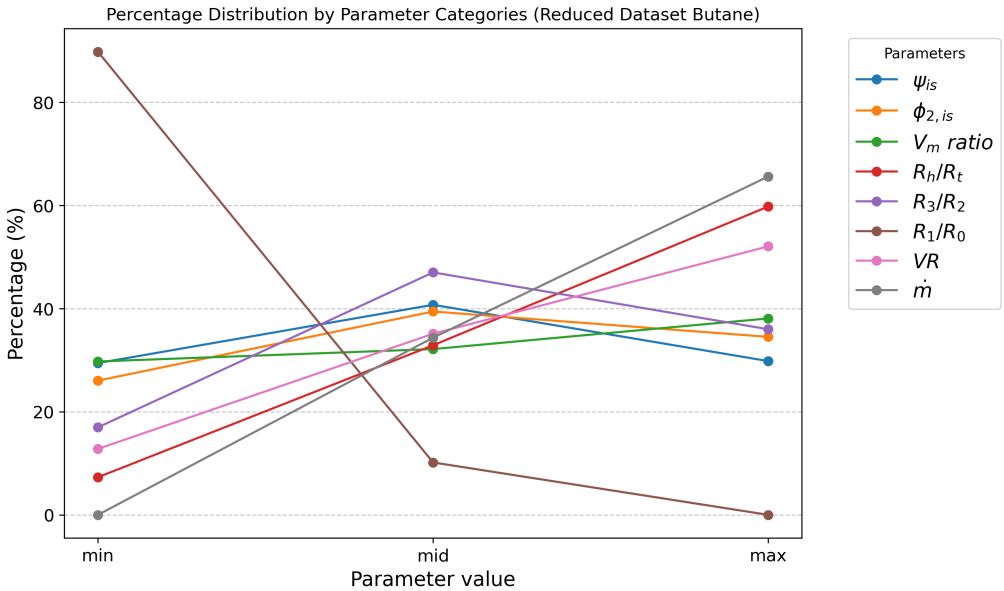
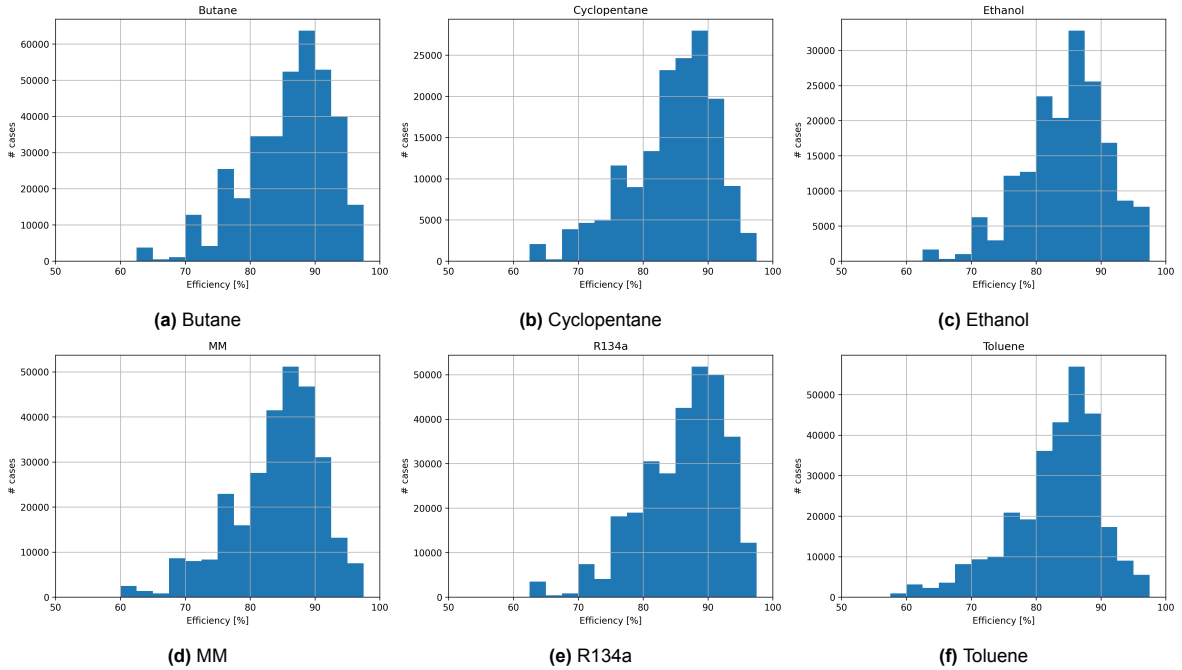


Figure 5.5: Distribution of parameters in reduced dataset for Butane

5.1.3. Efficiency and Weight Distribution

To gain a deeper understanding of how the surrogate model is trained and the predictions it is likely to produce, the efficiency and weight distributions of the dataset are analyzed. By identifying peaks in these distributions, it is possible to evaluate whether the model aligns with the most frequently occurring values. When certain efficiency or weight values appear more frequently in the dataset, the surrogate model can have a bias toward this particular part of the dataset. On the other hand, in regions where data is sparse, the model may exhibit greater uncertainty and tend to predict values corresponding to the dominant peaks. Working fluids with a larger range in turbine assembly weights are likely to produce a higher Mean Squared Error (MSE) compared to fluids with a smaller range, as the error can be more significant. The MSE is calculated over the complete dataset, whereas squared error can be used to evaluate the accuracy of individual samples.

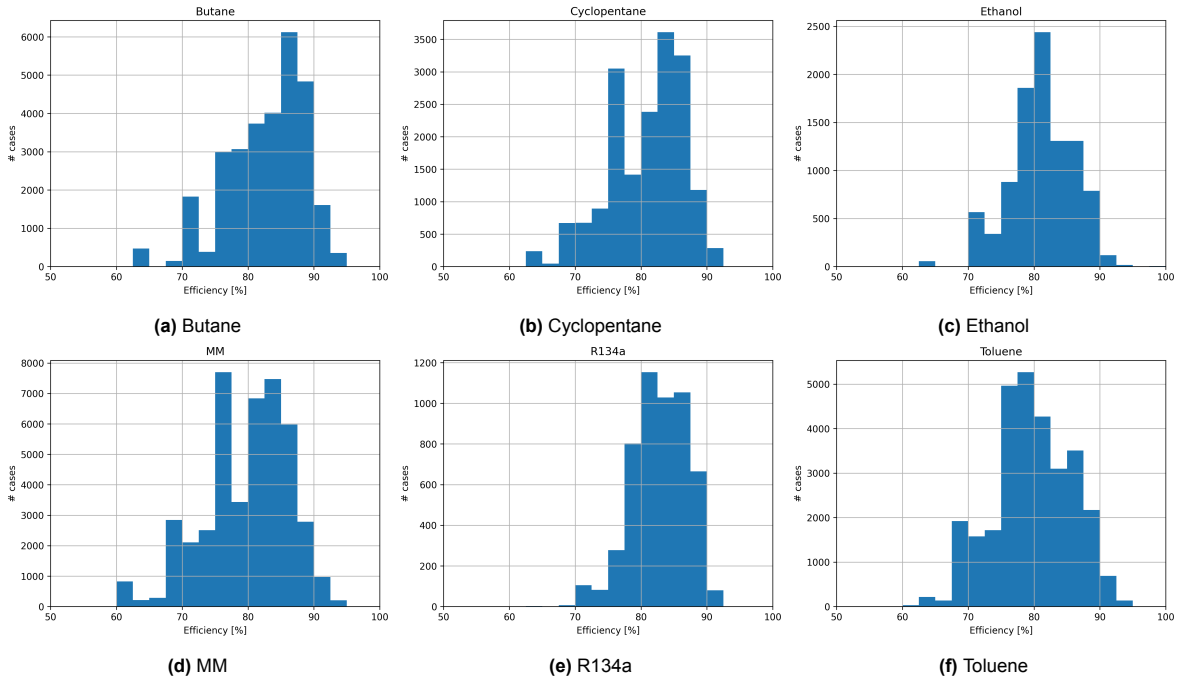
For the complete dataset, shown in Figure 5.6, the most frequently occurring efficiency values are found around $\eta = 87.5\%$ for working fluids butane, cyclopentane and R134a. Ethanol, MM and toluene, however, have a peak at $\eta = 85\%$. When filtering out the unfeasible designs by applying the constraints, as shown in Figure 5.7, the distribution changes. The data becomes more concentrated around the mean value and it contains a lower amount of outliers. The mean efficiency is presented in Table 5.4, where it can be seen that the mean efficiency will reduce when applying the feasibility constraints. In both the complete and reduced datasets, the mean efficiency values are very close together, suggesting that the working fluid has little impact on the efficiency of the turbine designs.



(a) Butane **(b)** Cyclopentane **(c)** Ethanol

(d) MM **(e)** R134a **(f)** Toluene

Figure 5.6: Efficiency histograms for complete dataset



(a) Butane **(b)** Cyclopentane **(c)** Ethanol

(d) MM **(e)** R134a **(f)** Toluene

Figure 5.7: Efficiency histograms for reduced dataset

Table 5.4: Mean efficiency of complete and reduced datasets for all investigated working fluids

Working fluid	Mean Efficiency [%]	
	Complete dataset	Reduced dataset
Ethanol	84.56	81.09
R134a	86.15	82.82
Butane	85.88	82.64
Cyclopentane	84.46	80.83
Toluene	82.97	79.68
MM	84.08	79.77

The majority of the turbine weights are concentrated at the lower values, although significant variations are observed between the different working fluids. Turbines designed for working fluids such as MM, ethanol and toluene exhibit extremely high weights, as shown in Table 5.5. This can be due to the higher upper limit of the volumetric flow ratio for these fluids. A higher *VR* can lead to an increased impeller radius R_2, which is shown in Figure 5.8. As a result, the overall turbine weight increases. Therefore, the choice of working fluid plays a key role in determining the turbine's weight characteristics.

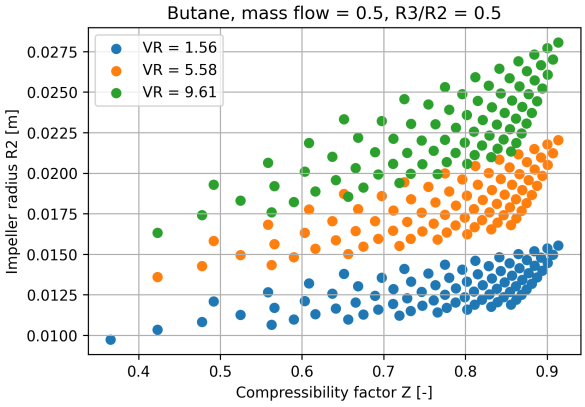


Figure 5.8: Impeller radius vs compressibility factor for the complete range of VR, while the DVs are fixed at their respecity mean value and the mass flow rate equals 0.5 kg/s

Despite these variations, most cases fall below 100 kg. The minimum weight predicted by TurboSim across all working fluids is approximately 0.02 kg, while the feasible designs have a minimum weight of around 1.5 kg. Figure 5.9 and 5.10 illustrate the weight breakdown of a light and heavy turbine operating with ethanol. The relative contribution of turbine components shifts with increasing weight. In small turbines, the heaviest components are the backplate (67.8%) and shroud casing (20.7%), while the impeller accounts for only 5.7% of the total turbine weight. This distribution changes for larger turbines, as seen in Figure 5.10, where an 86 kg turbine has the impeller as the dominant contributor (52.6%), with the backplate and shroud casing making up most of the remainder. The locking ring has a negligible effect and the stator blades contribute less than 1% of the total weight.

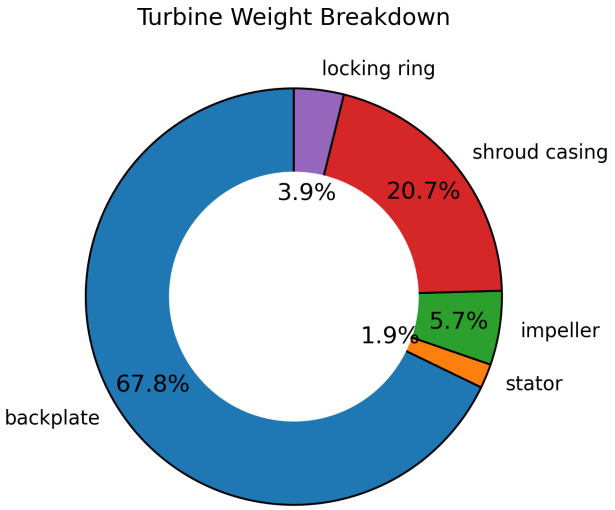


Figure 5.9: Turbine weight breakdown of a 0.128 kg turbine operating on ethanol

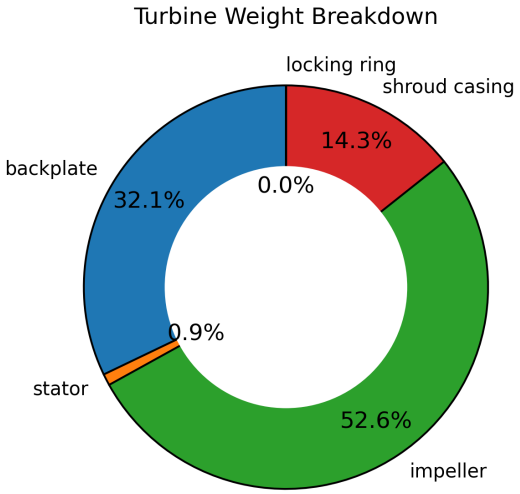


Figure 5.10: Turbine weight breakdown of a 86.009 kg turbine operating on ethanol

After applying the constraints, most extreme cases were filtered out for most fluids. However, for cyclopentane and toluene, which are the two more complex hydrocarbons, the data filtering primarily removes low weight designs while leaving the heavier cases unaffected, as can be seen in Table 5.5. The new mean weights are found in Table 5.6.

Table 5.5: Weight distribution for the complete and reduced datasets

Working fluid	Complete dataset		Reduced dataset	
	Minimum weight [kg]	Maximum weight [kg]	Minimum weight [kg]	Maximum weight [kg]
Ethanol	0.02	1344.26	1.60	741.32
R134a	0.01	11.31	1.60	7.89
Butane	0.02	37.48	1.53	25.10
Cyclopentane	0.02	94.14	1.58	94.14
Toluene	0.03	1268.48	1.54	1268.48
MM	0.03	730.18	1.60	478.40

Table 5.6: Mean weight of complete and reduced datasets for all investigated working fluids

Working fluid	**Mean Weight** [kg]	
	Complete dataset	Reduced dataset
Ethanol	21.61	55.68
Butane	1.60	5.51
Cyclopentane	4.36	14.35
R134a	0.64	3.10
Toluene	40.81	83.61
MM	16.58	40.91

5.2. Surrogate Models

Surrogate models have been generated to predict the efficiency and weight of radial-inflow turbines. In a first phase, a single surrogate model with two output nodes (i.e., the two quantity of interest, efficiency and weight) was trained using the data obtained from TurboSim simulations. These models will be referred to as Symbolic Regression 2 Outputs (SR2O). Subsequently, two additional surrogates, each one trained to predict a single quantity of interest (either the efficiency or the weight) and thus featuring a single output node, were generated using the same dataset. They will be abbreviated to SR1E and SR1W, respectively. The PySR [9] implementation of a symbolic regression model was chosen to generate the surrogates presented in this work, which was discussed in subsection 4.3.4. The present section provides a summary of the main results obtained after the training process, along with an assessment of the accuracy of the surrogate on a test dataset, performed using the Mean-Squared-Error (MSE) and the R-squared (R^2) performance metrics. This was done for both the surrogate model including two output nodes (SR2O) and the two additional models, each consisting of a single output node to predict each quantity of interest separately (SR1E and SR1W). The MSE quantifies the average squared error between actual values and predictions. A high MSE indicates that the model predictions deviate significantly from the actual values, which may be due to larger overall errors or due to the presence of outliers in the dataset. The R^2 score indicates the fit of the predictions compared to the actual data. A perfect fit would result in $R^2 = 1$, therefore, a high value is preferred. A more detailed explanation can be found in Appendix A.

Initially, all regression equations were limited to a maximum complexity (setting 'maxsize' in PySR) of 40. However, this limit was later increased to 60 to allow for higher R^2 values, which in turn give a better approximation of the trend in the generated data. The binary operators +, -, * and / were always included. The original decision was to include unique combinations of the binary operator *power* (^) and unary operators *sin, cos, tan, ln* and *exp*. However, it was found that in some cases the predicted weight was negative. This can be due to the fact that the datasets have the highest concentration of data located at the lower weights, as can be seen by the low mean values listed in Table 5.6, as well as, being a result of not filtering out the non-physical values in the loss models. To solve this issue, these non-physical designs were filtered out and additionally the unary operator *abs* (absolute value) was added to the list of operators to train SR1W models using the complete dataset. It was not added to the SR2O and SR1E models because it would be redundant in the efficiency expressions since the

predictions were always positive. Since unary operators are used, it was necessary to add constraints to avoid creating very complex expressions. These constraints were explained in subsection 4.3.4.

The dataset sizes are summarized in Table 5.7. Approximately 90% of the dataset was randomly selected and used to train the model (training set), while the remaining 10% was reserved for evaluating the R^2 (test set), which will be referred to as 'unseen' or 'new data'. An equal number of data points were randomly selected from the training dataset to compute the R^2 value for 'seen data'.

It was observed that the R^2 values of both cases (seen and new data) were nearly identical. In addition, this is also observed when comparing the average R^2 values of the different models that were tested in the verification phase, see Appendix D. Thus, indicating that a train-test split of 90/10 is sufficient to reach acceptable accuracies on unseen data. The selection of the best model was based on the highest R^2 score, this either being the one for 'seen data' or 'new data'.

Table 5.7: Datasets split between used for training and used for verification

Working fluid	**Complete dataset** [cases]		**Reduced dataset** [cases]	
	Training	Test	Training	Test
Ethanol	155 006	17 500	8691	1000
R134a	273 696	30 500	4757	500
Butane	322 525	36 000	26 562	3000
Cyclopentane	141 762	16 000	15 693	2000
Toluene	261 645	29 000	26 685	3000
MM	258 724	28 500	39 662	4500

5.2.1. Training Efficiency and Weight Simultaneously

When training a single model to predict both efficiency and weight simultaneously (SR2O), the model accuracy when predicting the two outputs can differ significantly. In particular, it was observed that a more complex model, indicated with a high complexity, is required to obtain a sufficiently accurate evaluation of the efficiency. In contrast, weight predictions can be satisfactory even with a simpler structure of the surrogate. However, the R^2 values for efficiency were consistently higher than those for weight, suggesting that PySR was able to identify clearer trends in the dataset for efficiency. Another key observation was that different mathematical operators were preferred for efficiency and weight. Although a possible alternative approach could be to include all possible operators and let the optimizer determine the best surrogate, it is generally preferable to minimize the number of operators [3].

Table 5.8 shows the best surrogate models to predict either of the outputs (efficiency and weight) for butane as working fluid. The corresponding R^2 values and complexities are shown. The best performing model at predicting the efficiency using the reduced dataset reaches a $R^2 = 0.8897$. Conversely, the weight prediction results in $R^2 = 0.5264$. This suggests that a trade-off exist for the selection of a suitable SR2O model.

Table 5.8: Best result for efficiency and weight when trained in a single model (SR2O) for working fluid butane, maximum allowed complexity is 40

Type dataset	Best performing on	R^2	Complexity	Selected operators	Corresponding parameter	R^2	Complexity
Reduced	Efficiency	0.8897	35	+, -, *, / , cos, tan, exp, only self-nesting allowed	Weight	0.5264	16
	Weight	0.7032	28	+, -, *, /, sin, exp, no nesting allowed	Efficiency	0.8189	12
Complete	Efficiency	0.9357	31	+, -, *, /, sin, tan, self nesting allowed	Weight	0.7504	19
	Weight	0.8329	24	+, -, *, /, ^, tan, exp, no nesting allowed	Efficiency	0.8382	14

5.2.2. Training Efficiency and Weight Separately

From the results listed in Table 5.8, one can see that the operators preferred by the PySR optimizer for generating surrogates with high scores in efficiency and weight prediction are vastly different. To minimize the number of operators, keep surrogate complexity at a minimum, and obtain an overall better fit of the data, the training process was carried out by feeding the algorithm with data of the efficiency and weight separately, resulting in the creation of two independent surrogates, SR1E and SR1W. Another benefit is that the two models can be trained in parallel on the Delft Blue Supercomputer, reducing the time spend on training the surrogate models.

An overview of the best performing models are listed in Table 5.9 and 5.10. It can be seen that increasing the maximum allowed complexity from 40 to 60 significantly increased the training time, even when the training was performed on the Delft Blue Supercomputer on 36 processing cores. Note that only the cases with the highest R^2 value are presented, the average values of the all training tests are summarized in Appendix D. In most cases, increasing the maximum complexity (maxsize) to 60 led to higher R^2 value. If this was not the case, it will be indicated in blue. The models that obtained a low R^2 value, indicating that they did not capture the trend of the dataset, are indicated in red.

Table 5.9: Overview of the best surrogate models for each type of trained model for working fluids butane, cyclopentane and ethanol

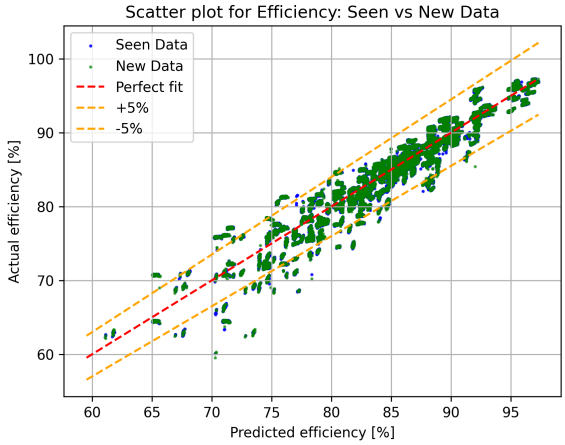
Fluid	Trained for	Type dataset	Maxsize	Best R^2	MSE	Equation complexity	Training time [s]
Butane	Efficiency	Reduced	40	0.890	4.066	27	138.1
			60	0.933	2.454	38	293.7
		Complete	40	0.910	4.058	33	388.6
			60	0.918	3.613	35	667
	Weight	Reduced	40	0.711	2.705	29	202.8
			60	0.706	2.784	41	378.2
		Complete	40	0.828	0.742	27	369.3
			60	0.857	0.635	42	2412.8
Cyclopentane	Efficiency	Reduced	40	0.889	3.576	39	151.8
			60	0.914	2.780	51	283.2
		Complete	40	0.909	4.143	25	268.0
			60	0.912	3.992	35	410.9
	Weight	Reduced	40	0.808	23.489	34	202.2
			60	0.841	21.049	30	226.6
		Complete	40	0.878	6.246	26	319.8
			60	0.892	5.570	32	315.9
Ethanol	Efficiency	Reduced	40	0.887	3.099	23	179.5
			60	0.900	2.626	37	266.1
		Complete	40	0.903	4.100	21	312.5
			60	0.908	3.880	39	332.9
	Weight	Reduced	40	0.928	517.0	27	185.3
			60	0.938	463.38	34	219.8
		Complete	40	0.887	380.22	31	216.0
			60	0.892	375.02	26	346.7

Table 5.10: Overview of the best surrogate models for each type of trained model for working fluids MM, R134a and toluene

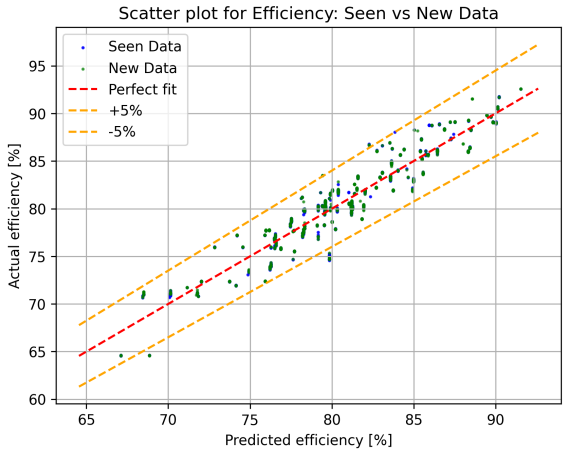
Fluid	Trained for	Type dataset	Maxsize	Best R^2	MSE	Equation complexity	Training time [s]
MM	Efficiency	Reduced	40	0.913	3.672	28	187.1
			60	0.937	2.855	35	376.6
		Complete	40	0.905	4.449	24	295.8
			60	0.938	2.934	39	439.1
	Weight	Reduced	40	0.868	308.818	27	209.5
			60	0.885	278.65	35	405.9
		Complete	40	0.886	154.342	33	397.9
			60	0.877	137.633	55	539.7
R134a	Efficiency	Reduced	40	0.920	1.720	28	240.3
			60	0.916	1.791	30	211.8
		Complete	40	0.913	3.732	28	350.9
			60	0.923	3.283	31	427.3
	Weight	Reduced	40	0.513	0.361	32	198.2
			60	0.569	0.351	30	222.6
		Complete	40	0.843	0.081	36	1179.2
			60	0.892	0.057	44	492.2
Toluene	Efficiency	Reduced	40	0.905	3.264	38	224.6
			60	0.909	3.143	52	265.4
		Complete	40	0.895	5.241	22	357.9
			60	0.910	4.391	40	602.1
	Weight	Reduced	40	0.926	1287.78	30	156.2
			60	0.931	1185.2	44	353.8
		Complete	40	0.867	901.884	30	317.1
			60	0.905	629.918	47	1124.4

The SR1E models were be trained on both the complete dataset and reduced dataset (feasible designs only) for each working fluid. For all fluids, the R^2 value exceeds 0.9 when trained on both datasets. The best and worst performing models are corresponding to working fluids siloxane MM and ethanol (alcohol), respectively. The R^2 score of the hydrocarbons tends to decrease with increasing molecular complexity, a trend observed for both the complete and reduced dataset. The complexity of the surrogate models trained on the complete dataset is between 35 and 40, with the exception of working fluid R134a, which has a complexity of 31.

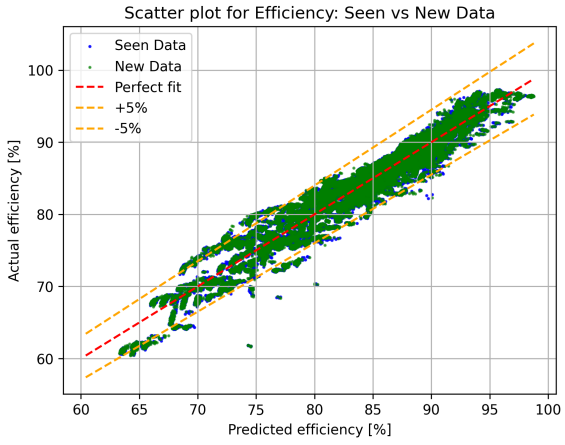
A visual representation of the R^2 values is provided in Figure 5.11, where both the 'seen' and 'new' datasets are shown. To illustrate the range of performance, only the fluids with the highest (MM) and lowest (ethanol) R^2 value are plotted. The results indicate that the testing cases align closely with the red line indicating a perfect fit ($y = x$). The efficiency datasets do not exhibit significant variations, as was illustrated in Figure 5.6 and 5.7. This is also confirmed by the relatively low MSE values compared to those of the weight models, with most surrogate predictions falling within a $\pm 5\%$ deviation from the actual values.



(a) Working fluid ethanol trained on the complete dataset ($155\,006$ cases), $17\,500$ cases were used to calculate the R^2 value ($R^2 = 0.908$)



(b) Working fluid ethanol trained on the reduced dataset (8691 cases), 1000 cases were used to calculate the R^2 value ($R^2 = 0.900$)



(c) Working fluid MM trained on the complete dataset ($258\,724$ cases), $28\,500$ cases were used to calculate the R^2 value ($R^2 = 0.938$)

(d) Working fluid MM trained on the reduced dataset ($39\,662$ cases), 4500 cases were used to calculate the R^2 value ($R^2 = 0.937$)

Figure 5.11: Comparison of efficiency predictions vs actual values for the best and worst performing working fluids using the complete and reduced datasets during training

The accuracy of the weight models (SR1W) show a relation with the working fluid properties. The R^2 scores of the SR1W models are in general slightly lower compared to the SR1E results. However, their R^2 values remain between 0.86 and 0.91, indicating that the overall trend in the data was successfully captured. Note that the weight dataset is heavily skewed toward lower weight values, leading to an

uneven data distribution. This is the case when considering ethanol, MM and toluene, as working fluids, for which the turbine weight is generally higher due to the larger range in investigated volumetric flow ratios, as tabulated in Table 5.5. The high MSE values for these fluids are a direct result of this large variation in turbine weight.

The reduced dataset for working fluid R134a contains the lowest number of cases. It consists of only 5257 cases in total, of which 4747 cases were used for training and 500 cases for evaluating the R^2 value of the surrogate model. The resulting $R^2 = 0.569$ indicates that the model did not manage to capture the trend in the dataset. This can be seen in Figure 5.12, where most of data points are located between $\pm$ 30% lines (orange). Figure 5.13 shows the change in the predicted weight distribution, shown as a histogram. It can clearly be seen that the reduced dataset has a different shape compared to the one using the surrogate model.

However, despite this, the turbine assembly weight itself exhibit very little variation (1.60 - 7.89 kg), which is reflected in its low MSE value (MSE = 0.35). Therefore, it can be concluded that the design space for R134a might not be appropriate. Relaxing the most violated constraints, such as the minimum thickness, blade height and radius constraints, can result in more feasible designs. Hence, a more appropriate dataset. However, as mentioned before, the design space of the dataset is not ideal, so relaxing the constraints has a minimal effect on it.

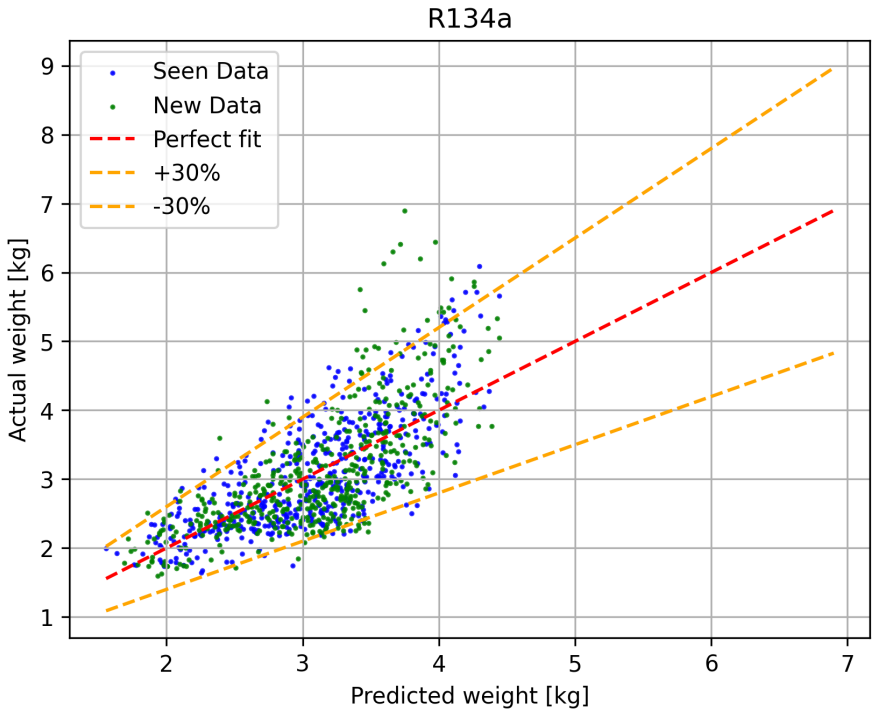
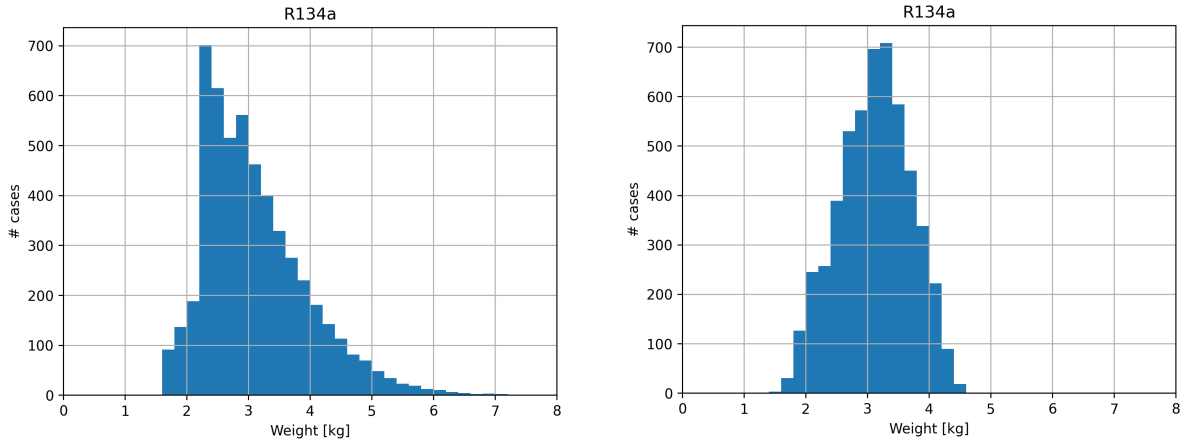


Figure 5.12: Working fluid R134a trained on the reduced dataset (4747 cases), 500 cases were used to calculate the R^2 value ($R^2 = 0.569$)



(a) R134a TurboSim

(b) R134a Surrogate model ($R^2 = 0.569$)

Figure 5.13: Weight histograms of the reduced dataset of working fluid R134a

On the contrary, the models for working fluids ethanol and toluene successfully captured the trend in the weight dataset, achieving R^2 values above 0.93, despite the datasets being widely spread. It can be seen that a higher number of predictions for ethanol (Figure 5.14) are closer to the perfect fit line compared to R134a (Figure 5.12), where the datapoints were more spread. However, the accuracy of the weight predictions for ethanol can be more than 30% off for turbine weights lower than 200 kg, as is indicated by the orange lines. The surrogate model has more datapoint underneath the red line, meaning that it has a tendency to overpredict the turbine weight.

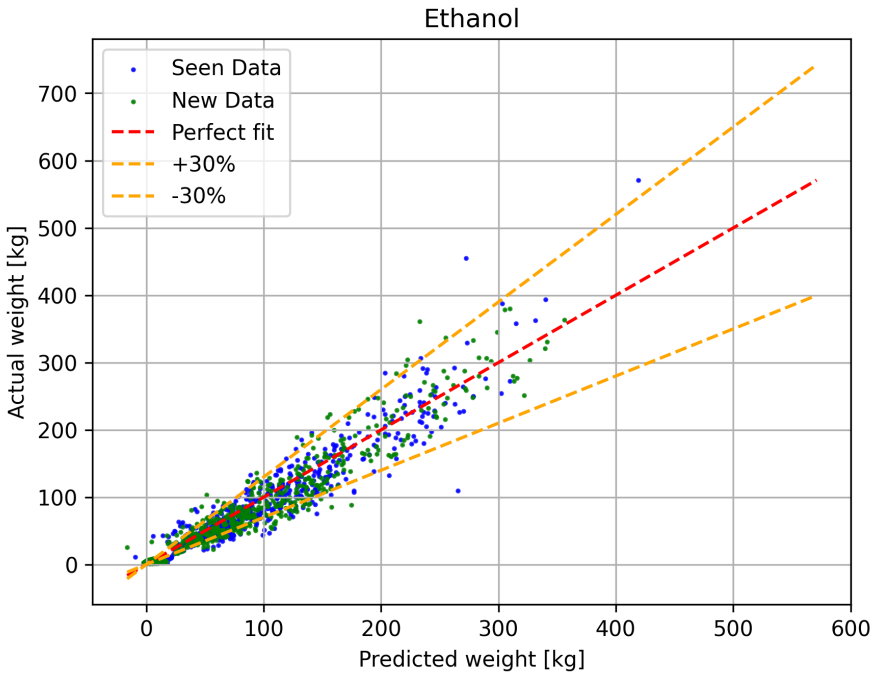


Figure 5.14: Working fluid ethanol trained on the reduced dataset (8691 cases), 1000 cases were used to calculate the R^2 value ($R^2 = 0.938$)

Accuracy of surrogate models based on DVs

The next step is to analyze the performance of the surrogate models in greater detail. The parametric study demonstrated that some design variables have a more significant impact on turbine efficiency and weight than others. The dataset contains three discrete values for the mass flow rate, the *VR*, as

well as for the six DVs. This approach significantly reduced the dataset size, making data generation more efficient. However, it also resulted in the loss of detailed information on how individual design variables influence efficiency and weight predictions.

The ability of the surrogate models to predict the trend of the training data will be examined in the following section. The predictions will be compared to the training dataset (three discrete values) as well as using a larger dataset (15 discrete values). To illustrate the difference between a well performing and a poorly performing model, working fluid R134a is used. There is a large difference in R^2 value for the SR1W models using working R134a and the other working fluids, as tabulated in Table 5.9 and 5.10. When trained on the complete dataset, the surrogate model for turbine weight achieved a high R^2 score of 0.892, capturing the overall trend well. However, when trained on the reduced dataset, the model failed to recognize any clear tends in the data, resulting in a much lower R^2 score of 0.569. In contrast, the efficiency model performed well in both cases, achieving R^2 scores of 0.923 and 0.926 for the complete and reduced datasets, respectively. Figure 5.15 to 5.18 show the surrogate predictions compared to the actual dataset for the most and least influential DVs used in the data generation. They show only a fraction of the dataset used during training. The process of selecting the settings for those figures as well as analyzing the presented results will be discussed in the following section.

Test settings
Identifying an ideal test case is challenging because the models are trained on multiple varying parameters. To address this, the analysis focuses on the regions in the dataset with the highest data density. The goal is to evaluate the model accuracy when varying a single design variable, this being either the one that has the largest impact or the smallest impact on efficiency and weight.

All input parameters, compressibility factor (Z), mass flow rate ($\dot{m}$) and volumetric flow ratio (*VR*), were fixed at the values for which the highest data density was identified. The compressibility factor is calculated for the maximum reduced temperature of 1.216 and a reduced pressure of 0.972 (highest subcritical investigated value). Both mass flow rate and *VR* are fixed at their mean value of 3.0 kg/s and 2.76, respectively, as they are evenly distributed across the complete dataset. Although the reduced dataset contains the highest number of cases operating at their respective maximum values, this analysis primarily focuses on how the surrogate trained on the complete dataset performs. Finding exact test points that are also included in the reduced dataset is more complex, but it can still be used to evaluate how well the model generalize outside its training range.

One design variable is allowed to vary over its complete range (three discrete points), while the remaining DVs are fixed at their respective mean values. Although this only covers three points in the large dataset, the surrogate models operate in a higher-dimensional space, making a complete visualization of the results challenging. Only the DVs which have the greatest and lowest impact on efficiency and weight are shown in the report for conciseness.

Analysis SR1W
Design variable R_3/R_2 has the highest influence on turbine efficiency as well as the turbine weight. The SR1W model using the complete dataset achieved a R^2 value of 0.892, which indicates that the surrogate model managed to capture the overall trend in the dataset. In contrast, the model trained on feasible designs only (reduced dataset) achieved a much lower R^2 value of 0.569, meaning that it failed to capture the trend. This discrepancy is illustrated in Figure 5.15. The points included in the complete dataset generated by TurboSim are indicated in red. The blue points were excluded from the training datasets, however, they are included as test points in the plot to show the trend of the weight generated by TurboSim. It can be seen that both surrogate models captured the shape of the data. The three points included in the complete dataset were removed in the reduced dataset because they did not comply with the constraints. As a result, the model trained on feasible designs predicts a much higher weight than the actual one, since the mean feasible weight increased, as was discussed in subsection 5.1.3.

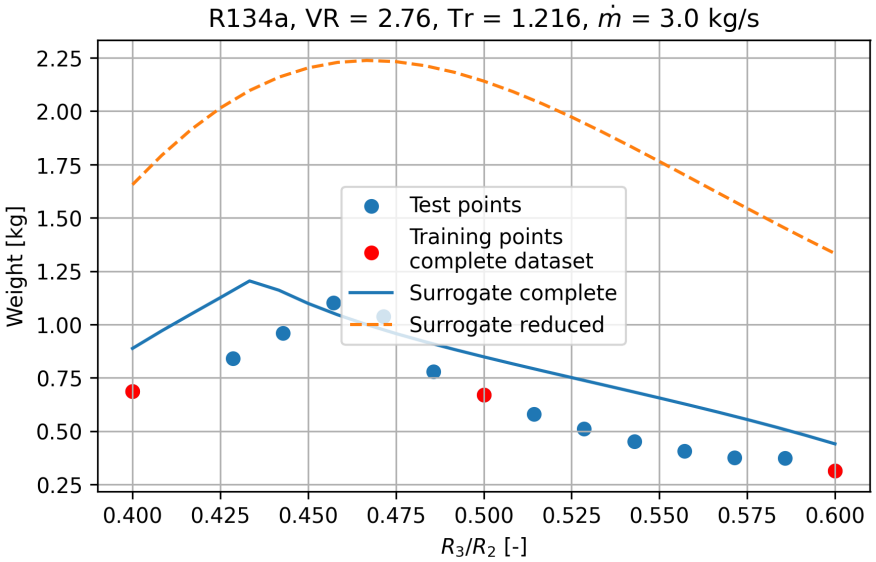


Figure 5.15: Weight predictions when varying design variable R_3/R_2, while keeping the others fixed at their mean value

Similarly, the surrogate model using only feasible designs for R134a cannot predict the weight accurately for the design variable with the lowest impact on weight. This is illustrated in Figure 5.16, where the variation in weight is significantly smaller compared to the cases where R_3/R_2 was varied. Once again, the three data points indicated in the plot are not included in the reduced dataset. As a result, the corresponding surrogate model was not able to reproduce the slight increase in the weight observed with increasing ϕ_{is}, further emphasizing the limitations of using the reduced dataset.

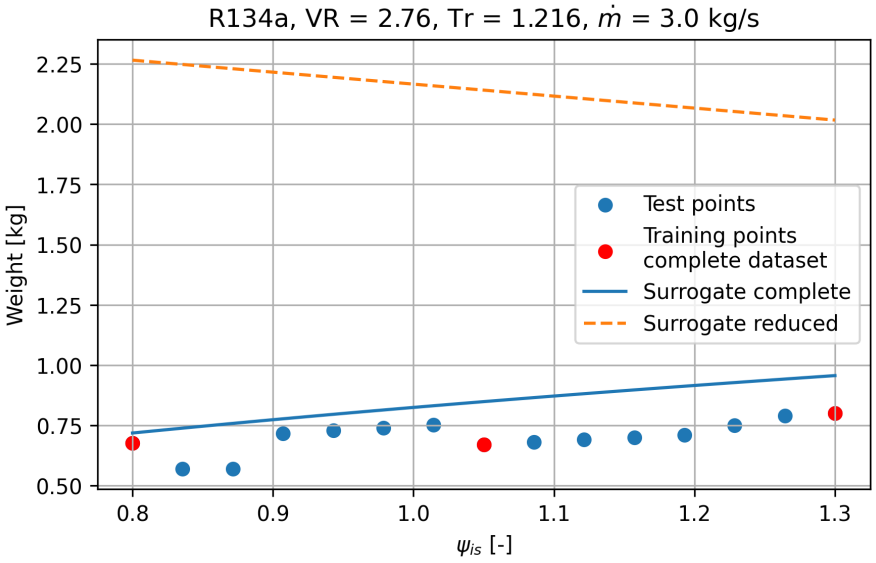


Figure 5.16: Weight predictions when varying design variable ψ_{is}, while keeping the others fixed at their mean value

Analysis SR1E

The efficiency predictions from the surrogate model trained on the complete dataset closely match the actual TurboSim results, as can be seen in Figure 5.17 and 5.18. However, the reduced surrogate model shows an unexpected behavior when varying R_3/R_2. This is likely the result of a scarce dataset. Since DV R_3/R_2 has the largest impact on efficiency, this is more prominent in Figure 5.18. This further confirms that surrogate models are a good tool to get an initial estimate of the turbine efficiency,

however, it is advised to run TurboSim to obtain the correct turbine efficiencies.

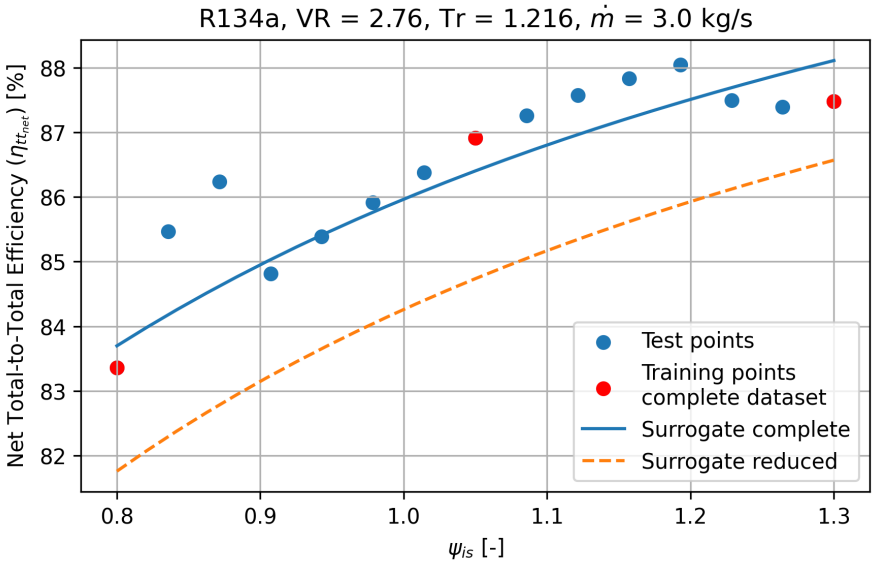


Figure 5.17: Efficiency predictions when varying design variable ψ_{is}, while keeping the others fixed at their mean value

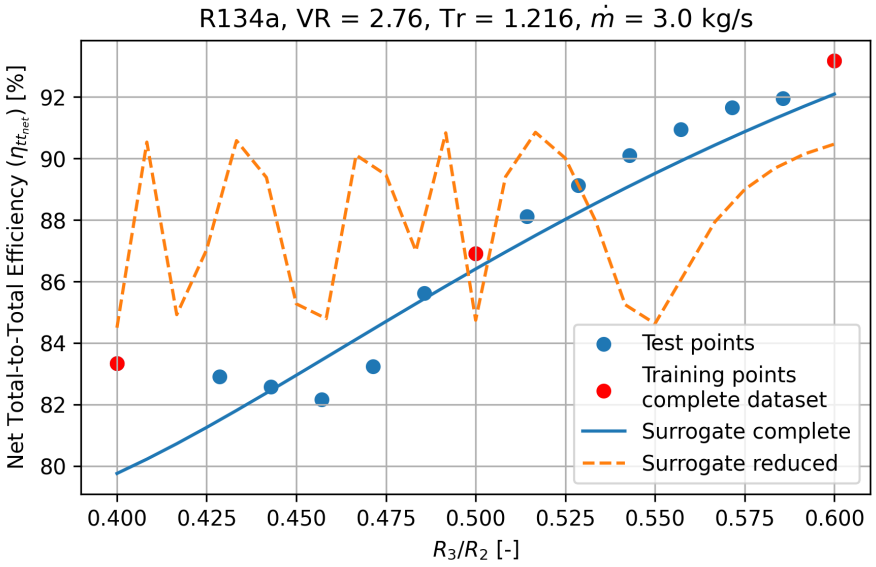


Figure 5.18: Efficiency predictions when varying design variable R_3/R_2, while keeping the others fixed at their mean value

Improving weight prediction accuracy

The accuracy of the surrogate model's weight predictions varies significantly. One reason for this larger discrepancy, compared to the efficiency predictions, is that the DVs have a larger impact on the weight. To investigate this, a test was performed using toluene, as it has the highest R^2 score, as well as the highest MSE among all working fluids trained on the complete dataset. The data was generated using DVs that have a comparable impact on weight as the six most influential DVs for efficiency. These are: $L_{ax}/\Delta R$ (39.55%), R_2/R_1 (38.33%), α_3 (25.23%), $(g/h)_{le}$ (12.80%), $(t/s)_{rot}$ (3.74%) and $(g/h)_{te}$ (2.42%). The results show that the R^2 score increased significantly from 0.905 to 0.936, while the MSE decreased from 629.9 to 437.0. This suggests that the model is more accurate in predicting weight.

6

Conclusion

This thesis documents research work on developing surrogate models to predict the efficiency and weight of ORC turbines for combined cycle engines. The objective of this research is to create a computationally efficient and highly accurate surrogate model based on symbolic regression trained to predict net total-to-total efficiency and weight of radial-inflow turbines. A parametric study was performed to identify the design variables (DVs) which have the highest influence on turbine efficiency and weight. The DVs investigated were mostly geometrical parameters, which have the highest influence on the turbine weight. These models were trained on two types of datasets: the complete dataset, which contains the results directly generated from TurboSim; and the reduced dataset which was obtained by applying feasibility constraints to the complete dataset. The main conclusions found in this study are presented in section 6.1, the limitations of the study and recommendations for further research are presented in section 6.2.

6.1. Main Conclusions

Based on the research performed and documented in this report, the following conclusions can be drawn:

1. The performance of ORC radial-inflow turbines can be predicted by using TurboSim, which is a tool developed by Majer and Pini [34]. However, it is a computationally expensive tool when used for generating large amounts of data. This results in the first research question:

   Research Question 1

   Can a symbolic regression surrogate model be developed for predicting the efficiency and weight for ORC RIT?

   The goal of this thesis was to develop a surrogate model using symbolic regression, which is trained on data from multiple working fluids, to accurately predict ORC RIT efficiency and weight. To allow for a flexible model, a diverse set of fluids were selected: butane, cyclopentane and toluene (hydrocarbons); ethanol (an alcohol); R134a (a refrigerant) and siloxane MM.

   A single surrogate model trained on data of all fluids considered and based on four input parameters, that are, the molecular complexity, the compressibility factor, the mass flow rate and the volumetric flow ratio, was initially sought after. However, due to the widely different thermodynamic properties of the different fluids, separate surrogate models had to be developed for each working fluid.

   The Python package PySR was chosen as a development framework to generate surrogate models suitable to predict single output and multiple outputs from the training data. However, it was observed that selecting the best model was difficult because it only calculated the accuracy per output and no overall accuracy is computed. In such cases, a trade-off should be made to choose

the model that best fits its application. However, a possible solution to address the shortcomings of the multiple output surrogate model, is to generate two single output surrogate models for each working fluid considered. These can be trained independently on data generated by TurboSim (complete dataset), as well as, trained on only feasible designs (reduced dataset), which were obtained by applying constraints.

In conclusion, while multi-output symbolic regression was used to generate a surrogate able to predict the turbine efficiency and weight, it was observed that using separate surrogate models yields better overall fitting of the data, which comes at the cost of computational overhead to train twice as many machines for each working fluid considered. This procedure resulted in the generation of 24 different surrogate models.

2. Majer and Pini [34] studied the design guidelines for high-pressure ratio supersonic RIT of ORC systems. The authors managed to devise best practices for the selection of the turbine duty coefficients that maximize the efficiency, however turbine weight was not considered in the work. Krempus et al. [28] showed that the turbogenerator mass is about one-third of the power unit mass (total mass excluding the dry mass of the engines and the generator mass). This highlights the need to include the turbine weight into the design process of RIT ORC turbines. Since the surrogate model is designed to be flexible and capable of making predictions for multiple types of working fluids, it is important to examine how the choice of working fluid affects efficiency and weight. This leads to the following research question:

   Research Question 2

   What is the impact of working fluid on ORC RIT efficiency and weight?

   Turbine designs featuring net total-to-total efficiency ranging between 60% and 98% were obtained for all working fluids considered in the analysis. Fluids with lower molecular complexity achieved the highest mean efficiency. R134a has the highest mean efficiency in both datasets, 86.15% for complete dataset and 82.82% for the reduced dataset. In the complete dataset, butane follows closely with a mean efficiency of 85.88%. Ethanol, cyclopentane and MM have nearly identical mean efficiencies of 84.56%, 84.46% and 84.08% respectively, while toluene shows the lowest mean efficiency at 82.97%.

   Applying constraints to filter out the unfeasible designs, also removed the outliers, leading to a slight reduction in mean efficiency. As before, the highest efficiencies were found for fluids with the lowest molecular complexity. The new mean efficiencies, ranked from highest to lowest, are: R134a (82.82%), butane (82.64%), ethanol (81.09%), cyclopentane (80.93%), MM (79.77%) and toluene (79.68%).

   In contrast, the impact of the working fluid on ORC RIT weight is more pronounced. A clear relationship exists between the volumetric flow ratio (*VR*) and the turbine weight. The larger the investigated range, the higher the turbine assembly weights registered. Based on the dataset generated by TurboSim (complete dataset), the working fluids can be ranked in order of increasing mean turbine weight: R134a (0.64 kg), butane (1.60 kg), cyclopentane (4.36 kg), MM (16.58 kg), ethanol (21.61 kg) and toluene (40.81 kg).

   The applied constraints eliminated most of the turbine assemblies below 2 kg, since the minimum weight found in these datasets is around 1.5 kg. However, the ranking of the working fluids remained unchanged. Turbines operating with R134a result in the lightest designs, while those using toluene are the heaviest.

   To conclude, the working fluid has a minor effect on efficiency but a significant influence on turbine weight. Working fluid R134a appears to be the best choice, since it has the lowest mean turbine weight and the highest efficiency. However, the reduced dataset for R134a is highly constrained, it only contained 0.8% of the tested cases ($656\,100$ cases per working fluid). Alternatively, the largest dataset was found for butane: 54.6% of the tested cases found a solution in TurboSim and the reduced dataset contains a much larger fraction of the cases (8.2% compared to 1.7% for R134a).

3. The TurboSim model relies on numerous input variables, including fluid properties, operating conditions and 17 geometrical parameters. A sensitivity study was conducted to identify the most influential design variables, which has two main advantages. First, including all design variables in the training dataset for the surrogate model would result in a substantial increase in training time. Second, the objective of this thesis is to asses the feasibility of developing a highly accurate surrogate model, making it effective to begin with a smaller set of design variables. This parametric study allowed to answer the following research question:

   Research Question 3

   What are the design variables that have the largest impact on efficiency and weight?

   The parametric study was performed by varying a single design variable in 10 steps between its minimum and maximum values (stated in Table 4.3), while keeping the other DVs fixed at their average values. The relative change, expressed as a percentage increase, was calculated by dividing the difference between the maximum and minimum values by the minimum value and multiplying by 100.

   For efficiency, the six most influential parameters – ranked from highest to lowest – are R_3/R_2 (33.47%), R_h/R_t (17.86%), $L_{ax}/\Delta R$ (11.43%), $\phi_{2,is}$ (10.13%), ψ_{is} (9.53%) and $(g/h)_{le}$ (6.37%). Conversely, the parameters with the greatest impact on weight are R_3/R_2 (414.38%), R_h/R_t (371.77%), $\phi_{2,is}$ (91.77%), Vm ratio (80.68%), R_1/R_0 (73.03%) and ψ_{is} (48.27%).

   The DVs are mostly geometric parameters, which in return will strongly influence the turbine weight. The surrogate models are trained on the six DVs that have the highest impact. Since the initial goal was to train the models for efficiency and weight simultaneously, the six highest scoring design variables were selected, these are the ones for weight: R_3/R_2, R_h/R_t, $\phi_{2_{is}}$, V_m ratio, R_1/R_0 and ψ_{is}.

4. To evaluate the reliability of the surrogate model, it is essential to assess its accuracy and determine whether it varies depending on the selected working fluid. This lead to the following research question:

   Research Question 4

   What is the accuracy of the surrogate model and will it change with working fluid?

   The multiple output surrogate models generated to predict the efficiency and the weight simultaneously (SR2O) were found to yield insufficient accuracy, and were therefore replaced by two single output models, each one able to predict either the efficiency or the weight of the turbine, SR1E and SR1W respectively.

   These models were evaluated based on two complementary metrics, i.e., the Mean-Squared-Error (MSE) and R-squared (R^2).

   All efficiency models – either if trained on the complete dataset or on the reduced dataset – achieved R^2 values above 0.9, indicating overall good fit to the dataset. The maximum MSE was found to be around 4.39, indicating good overall accuracy. When MM was selected as the working fluid, the surrogate model achieved the highest R^2 value for both datasets, 0.938 and 0.937 for the complete and reduced dataset, respectively. Ethanol scored the lowest with a R^2 score of 0.908 for the complete dataset and 0.900 for the reduced dataset. This suggests that the training algorithm captures the trend in siloxane data most effectively, while performing worst for alcohols. The difference between the refrigerant R134a and the hydrocarbons is minimal, with the refrigerant achieving a R^2 of 0.923, close to MM. A trend can be observed among hydrocarbons trained on both datasets: butane scored the highest R^2, followed by cyclopentane and lastly toluene. This indicates that the algorithm more easily captures trends in the datasets of less complex fluids.

However, the accuracy of the models used to predict the weight was significantly lower in contrast to the efficiency. There is a large difference in weight datasets when comparing the different working fluids. As mentioned before, the range of generated weight is related to the investigated range of the volumetric flow ratio for every working fluid. The smallest range was observed for R134a (maximum 11.3 kg), while when using ethanol, MM and toluene as working fluids, turbine assembly weights above 100 kg were found.

The weight models trained on the complete dataset performed relatively well in detecting the trend in the dataset, because they have a R^2 score between 0.857 and 0.905. The best performing fluid is toluene although it has highest MSE value of all fluids. This indicates that the surrogate model was able to capture the trend in the dataset, despite it being widely spread. The model trained for working fluid butane performed the worst on capturing the trend, however, the MSE value is very low 0.63 compared to 629.92 for toluene. This shows that the spread of the data does not directly affect the ability to capture the trend in the dataset. However, it does have an affect on the accuracy of the predictions. Significantly, within the hydrocarbons, the most molecularly complex fluid (toluene) exhibited the highest R^2 score, while the least complex (butane) had the lowest, suggesting a relationship between molecular complexity and model performance for this fluid category.

Applying the constraints to the dataset removed mostly the outliers of the datasets but had little direct impact on most R^2 scores. While the R^2 value for ethanol and toluene increased, they decreased for the other fluids. The model for R134a performed the worst, with a R^2 score of 0.569, likely due to the small size of the reduced dataset – only 5257 points, representing just 0.8% of all the tested designs in TurboSim or 1.7% of the complete dataset.

The best performing model is the one for ethanol, which achieved a R^2 score of 0.938, however, it has a high score for MSE (463.4). This again suggests that even with widely spread data, it is still possible to develop a surrogate model that captures the trend well. The best performing fluid for capturing the trend is again ethanol, while the refrigerant R134a performed the worst. The same pattern was observed for hydrocarbons: the most complex fluid exhibited the highest trend accuracy, while the least complex performed the worst.

To conclude, the working fluid does not have a great affect on the accuracy of the models trained for predicting the turbine efficiency. However, the accuracy of the weight models are fluid dependent. The models trained on the complete dataset were able to capture the trends in the weight dataset well. However, the accuracy of the models using ethanol, MM and toluene can vary more compared to the other fluids because of their higher MSE values. When using the reduced datasets, it was not possible to capture the trend for R134a and only slightly for butane. The other fluids performed well.

6.2. Limitations and Recommendations for Future Work

This study presents several limitations in the modeling approaches used. In this section, these limitations are discussed, along with suggestions for future research to address them.

- The parametric study conducted in section 4.2 normalized only the results, not the design variable values. This can give a misleading impression of the influence of certain DVs on the turbine efficiency and weight. The DVs were normalized by dividing the difference between the minimum and maximum value by the mean value of the DV and multiplying it by 100. Although the top five most influential DVs remained unchanged, the normalization affected the ranking of the less influential parameters. The new top six most influential DVs – ranked from highest to lowest – are R_3/R_2 (10.36%), $\phi_{2,is}$ (3.84%), R_h/R_t (3.72%), R_1/R_0 (2.26%), R_2/R_1 (1.94%) and V_m ratio (1.61%). Since this study demonstrated that accurate surrogate models can be developed using a subset of six DVs, future work should consider extending the models to include additional DVs. A correctly normalized parametric study should be conducted to ensure the proper selection of the influential DVs, enabling the models to predict turbine efficiency and weight for a broader range of cases.

- The training datasets in this study were generated using TurboSim. However, an alternative approach could be to use optimized data, such that the model is only trained on optimized turbine architectures. This could bias the predictions toward the optimal Pareto front, however, it is important to note that the surrogate model's predictions would not necessarily represent optimal designs.

  A preliminary test using a Pymoo optimization revealed that obtaining a single optimized result took quite a long time (about 2h), making it computationally expensive. Further research could explore more efficient methods for generating optimized datasets and compare the impact of this approach on the surrogate model's predictive performance.

- TurboSim successfully generated results for only 24-55% of the investigated designs (Table 5.1). To increase the dataset size, while maintaining the number of investigated designs, the following strategies could be explored:

  1. *Adjusting TurboSim's termination criteria*: Relaxing the termination criteria in iterative loops could allow more cases to be computed. However, this may also increase the computation time.
  2. *Refining the design space*: Few cases in the dataset operate at the minimum value of the design variables flow ($\phi_{2,is}$) and R_3/R_2, shown in section C.1. Increasing their lower limits might result in a larger dataset.
  3. *Optimizing the volumetric flow ratio range*: For ethanol, most cases were obtained at the minimum VR value, indicating that the selected upper value might be too high. Narrowing the range of *VR* might lead to a better model.

- Since symbolic regression is a genetic algorithm, it does not naturally converge to a final expression, as it continuously explores new functional forms. In this thesis research, the number of iterations ('niterations') was limited to 40 to allow for shorter training times, while still achieving high accuracy.

  According to M. Cranmer, the developer of PySR, once all settings have been finetuned, the model should be trained using a significantly larger number of iterations, potentially running it for a week or until the job finishes by complying with the stopping criteria [3]. However, due to the 24 hour job limit on a student account on the Delft Blue Supercomputer used for this research, this was not feasible. Further research should evaluate the trade-off between extended training duration and model accuracy.

- The accuracy of the weight predictions varies significantly depending on the investigated case. A reason for this could be the inherent high variance in the dataset, which shows a large spread in turbine weights compared to the efficiency dataset. While PySR was able to identify trends in the data for most working fluids, except R134a for the complete dataset, the MSE of these models remains extremely high, indicating that the predictions are not highly accurate. This variability in the data could be due to the sensitivity of turbine weight to certain design variables, which leads to significant fluctuations in the results across similar cases. There are several approaches that can be taken to address this:

  1. *Refining the VR range*: It was observed that turbine weight tends to increase as the *VR* range investigated becomes larger. Lowering the maximum investigated *VR* could reduce the variability in turbine weights. Additionally, as shown in Figure C.1 to C.6, the number of cases in the datasets operating at the maximum *VR* is typically lower than those at the minimum or mean *VR*. Adjusting the *VR* ranges could not only reduce variability but also increase the number of cases in the dataset generated by TurboSim.
  2. *Denoising the training data*: PySR was applied directly to the raw training datasets without denoising, as each DV was only investigated at three data points. As seen in Figure 5.15 to 5.18, the three red training points do not capture the full dataset trend. PySR has a built in denoising tool, which could be tested to determine if it enhances prediction accuracy.

3. *Comparing Neural Networks*: An alternative approach is the use of Neural Networks, as was done by Giuffré et al. [15]. NNs can model complex, nonlinear relationships within the data, potentially improving prediction accuracy. However, NNs do not provide the same level of interpretability as symbolic regression models. To address this, PySR could be integrated into PyTorch as a layer, allowing for both accurate predictions and the extraction of symbolic expressions [9]. It is advised to start with a simple NN, e.g. Multi-layer perceptron (MLP). While NNs might offer higher predictive performance, they can come at a cost of reduced transparency, which might be critical in turbine design where understanding the relationships between variables is important.

- The surrogate models in this report are created for RITs, not turbogenerators. These models could, however, be adapted for different configurations. For example, the ORC system analyzed by Krempus et al. [28] (Figure 2.2) contains two generators in the main engine system and one in the ORC system. Further studies could explore the differences in performance between the following two configurations: one configuration could be a turbogenerator (turbine + generator), while another configuration could be to directly couple the turbine to one of the generators in the main engine, making it a direct drive system.

  Another potential application of these surrogate models is in determining which settings should be tested in an experimental setup. While the ORCHID test bench at the Aerospace Faculty is not yet operational, once available, it is expected to be in high demand. Therefore, efficient planning of experiments will be essential. Surrogate models could help by predicting expected outcomes beforehand. This would allow researchers to prioritize necessary tests, potentially reducing future operational time and costs.

- A test was conducted by running a single-objective optimization in Pymoo for the working fluid MM. When optimizing for efficiency, the results revealed a significant difference in computational efficiency: the surrogate model converged after 6 seconds, requiring 15 generations across 32 cores on the Delft Blue Supercomputer. In contrast, the TurboSim model took $54\,932$ seconds and 35 generations to complete the same task. A similar trend was observed when optimizing for weight, where the surrogate model converged in 11 seconds after 26 generations, while TurboSim required $78\,703$ seconds and 45 generations.

  The optimized efficiency predicted by both models were very similar. The optimized input parameters and design variables for both optimizations are presented in Table 6.1. While the design variables exhibit similar values, the input parameters, such as reduced pressure, volumetric flow ratio and mass flow rate, show a more significant discrepancy. Despite these differences, the advantages of using the surrogate model over TurboSim are evident. The surrogate model offers a significantly faster optimization process, requiring fewer generations, and provides results that are sufficiently accurate for obtaining optimized values.

  In contrast, the weight optimization produced unrealistic results. TurboSim predicted an optimal turbine weight of 1.67 kg, whereas the surrogate model yielded an infeasible value of $1.5 \cdot 10^{-8}$ kg. The main differences were observed in volumetric flow ratio and mass flow rate: the surrogate model optimized for a *VR* of 4.9 and a mass flow rate of 4.3 kg/s, while TurboSim required a significantly higher *VR* of 23.6 and a much lower mass flow rate of 0.6 kg/s. These inconsistencies confirms that the surrogate model's weight predictions are not yet reliable for practical application. While a constraint on minimum weight could be imposed, this would not resolve the underlying issue, as the optimization would simply seek configurations that meet the constraint rather than accurately predicting weight. Future work should focus on improving weight prediction by refining the training dataset, incorporating additional constraints, or exploring alternative modeling approaches.

Table 6.1: Pymoo optimization using TurboSim and the surrogate model for working fluid MM

DV	TurboSim	Surrogate	DV	TurboSim	Surrogate
ψ_{is}	1.24	1.26	$\dot{m}$	4.00	4.86
$\phi_{2,is}$	0.43	0.46	R_h/R_t	0.46	0.44
T_r	0.97	0.94	R_3/R_2	0.60	0.57
P_r	1.20	1.02	R_1/R_0	0.67	0.70
VR	12.13	9.05			

- The obtained analytical expressions are complex, and it is advised to remove (part of) the unary operators to obtain simpler functions. However, it was observed that the models trained on only binary operators were less accurate in capturing the trend of the dataset. This can be due to large number of variables used during training. A hyperparameter optimization was performed for the complete dataset, presented in section E.9, and it shows that different fluids prefer different settings. When expanding these surrogate models to include more DVs in training, one can use these tables to select the operators for training the models. This will reduced the number of training runs required.

References

[1] F. Alshammari, M. Elashmawy, and M. Bechir Ben Hamida. “Effects of working fluid type on powertrain performance and turbine design using experimental data of a 7.25ℓ heavy-duty diesel engine”. In: *Energy Conversion and Management* 231 (2021), p. 113828. ISSN: 0196-8904. DOI: https://doi.org/10.1016/j.enconman.2021.113828.

[2] J. Alzubi, A. Nayyar, and A. Kumar. “Machine Learning from Theory to Algorithms: An Overview”. In: *Journal of Physics: Conference Series* 1142.1 (Nov. 2018). Publisher: IOP Publishing, p. 012012. ISSN: 1742-6596. DOI: 10.1088/1742-6596/1142/1/012012.

[3] Astroautomata. *PySR: Symbolic Regression in Python*. 2025. URL: https://astroautomata.com/PySR/tuning/.

[4] A. Barzkar and M. Ghassemi. “Electric Power Systems in More and All Electric Aircraft: A Review”. In: *IEEE Access* 8 (2020), pp. 169314–169332. DOI: 10.1109/ACCESS.2020.3024168.

[5] Ian H. Bell et al. “Pure and Pseudo-pure Fluid Thermophysical Property Evaluation and the Open-Source Thermophysical Property Library CoolProp”. In: *Industrial & Engineering Chemistry Research* 53.6 (2014), pp. 2498–2508. DOI: 10.1021/ie4033999. eprint: http://pubs.acs.org/doi/pdf/10.1021/ie4033999. URL: http://pubs.acs.org/doi/abs/10.1021/ie4033999.

[6] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[7] S. L. Brunton and J. N. Kutz. *Data Driven Science & Engineering: Machine Learning, Dynamical Systems, and Control*. 2nd ed. Cambridge University Press, 2017.

[8] P. Colonna and A. Guardone. “Molecular interpretation of nonclassical gas dynamics of dense vapors under the van der Waals model”. In: *Physics of Fluids* 18.5 (May 2006), p. 056101. ISSN: 1070-6631. DOI: 10.1063/1.2196095. eprint: https://pubs.aip.org/aip/pof/article-pdf/doi/10.1063/1.2196095/15757031/056101_1_online.pdf. URL: https://doi.org/10.1063/1.2196095.

[9] M. Cranmer. *Interpretable Machine Learning for Science with PySR and SymbolicRegression.jl*. 2023. arXiv: 2305.01582 [astro-ph.IM].

[10] C. De Servi et al. “Exploratory assessment of a combined-cycle engine concept for aircraft propulsion”. In: *Proceedings of the 1st Global Power and Propulsion Forum: an 16-18*. Article GPPF-2017-78. Zurich, Switzerland, 2017.

[11] C. M. De Servi et al. “Design Method and Performance Prediction for Radial-Inflow Turbines of High-Temperature Mini-Organic Rankine Cycle Power Systems”. In: *Journal of Engineering for Gas Turbines and Power* 141.9 (Aug. 2019), p. 091021. ISSN: 0742-4795. DOI: 10.1115/1.4043973. eprint: https://asmedigitalcollection.asme.org/gasturbinespower/article-pdf/141/9/091021/6423587/gtp_141_09_091021.pdf. URL: https://doi.org/10.1115/1.4043973.

[12] S. N. Dhage and C. K. Raina. “A review on Machine Learning Techniques”. In: *International Journal on Recent and Innovation Trends in Computing and Communication (IJRITCC)* 4.3 (Mar. 2016), pp. 395–399.

[13] J. Faber and D. S. Lee. “Bridging the gap – the role of international shipping and aviation”. In: *Emissions Gap Report 2020*. United Nations, 2021, pp. 52–61.

[14] J. Ferreira, M. Pedemonte, and A. I. Torres. “A Genetic Programming Approach for Construction of Surrogate Models”. In: *Proceedings of the 9th International Conference on Foundations of Computer-Aided Process Design*. Ed. by S. Garcia Muñoz, C. D. Laird, and M. J. Realff. Vol. 47. Computer Aided Chemical Engineering. Elsevier, 2019, pp. 451–456. DOI: https://doi.org/10.1016/B978-0-12-818597-1.50072-2.

[15] A. Giuffré et al. “Data-driven modeling of high-speed centrifugal compressors for aircraft Environmental Control Systems”. In: *International Journal of Refrigeration* 151 (2023), pp. 354–369. ISSN: 0140-7007. DOI: https://doi.org/10.1016/j.ijrefrig.2023.03.019.

[16] A. J. Glassman. *Computer Program for Design Analysis of Radial-Inflow Turbines*. Tech. rep. TN D-8164. NASA, 1976.

[17] N. Gray et al. “Decarbonising ships, planes and trucks: An analysis of suitable low-carbon fuels for the maritime, aviation and haulage sectors”. In: *Advances in Applied Energy* 1 (2021), p. 100008. ISSN: 2666-7924. DOI: https://doi.org/10.1016/j.adapen.2021.100008.

[18] Alberto Guardone et al. “Nonideal Compressible Fluid Dynamics of Dense Vapors and Supercritical Fluids”. In: *Annual Review of Fluid Mechanics* 56.Volume 56, 2024 (2024), pp. 241–269. ISSN: 1545-4479. DOI: https://doi.org/10.1146/annurev-fluid-120720-033342. URL: https://www.annualreviews.org/content/journals/10.1146/annurev-fluid-120720-033342.

[19] V. Hamolia et al. “INTRUSION DETECTION IN COMPUTER NETWORKS USING LATENT SPACE REPRESENTATION AND MACHINE LEARNING”. In: *International Journal of Computing* 19.3 (Sept. 2020), pp. 442–448. DOI: 10.47839/ijc.19.3.1893.

[20] B. He et al. *Taylor Genetic Programming for Symbolic Regression*. 2022. arXiv: 2205.09751 [cs.NE].

[21] AJ Head, A Gangoli Rao, and F Yin. *Performance trends of high-bypass civil turbofans*. English. Netherlands: Delft University of Technology, 2015.

[22] CFM International. *CFM RISE PROGRAM - Revolutionary Innovation for Sustainable Engines*. [White paper]. 2021. URL: https://www.cfmaeroengines.com/wp-content/uploads/2021/07/CFM_RISE_Whitepaper_Media.pdf.

[23] O. Y. Al-Jarrah et al. “Efficient Machine Learning for Big Data: A Review”. In: *Big Data Research* 2.3 (2015). Big Data, Analytics, and High-Performance Computing, pp. 87–93. ISSN: 2214-5796. DOI: https://doi.org/10.1016/j.bdr.2015.04.001.

[24] A. Javanshir, N. Sarunac, and Z. Razzaghpanah. “Thermodynamic Analysis of ORC and Its Application for Waste Heat Recovery”. In: *Sustainability (Switzerland)* 9 (Oct. 2017). DOI: 10.3390/su9111974.

[25] B. Kanberoglu et al. “The effects of different working fluids on the performance characteristics of the Rankine and Brayton cycles”. In: *International Journal of Hydrogen Energy* 49 (2024), pp. 1059–1074. ISSN: 0360-3199. DOI: https://doi.org/10.1016/j.ijhydene.2023.10.058.

[26] A. H. Khan et al. *Digital Twin and Artificial Intelligence Incorporated With Surrogate Modeling for Hybrid and Sustainable Energy Systems*. 2022. arXiv: 2210.00073 [cs.AI].

[27] J. R. Koza. “Genetic programming as a means for programming computers by natural selection”. In: *Statistics and Computing* 4.2 (June 1, 1994), pp. 87–112. ISSN: 1573-1375. DOI: 10.1007/BF00175355.

[28] D. Krempus et al. “ORCWaste Heat Recovery System for the Turboshaft Engines of Turboelectric Aircraft”. English. In: *PRoceedings of the Aerospace Europe Conference 2023 10th EUCASS 9th CEAS*. Aerospace Europe Conference 2023 : Joint 10th EUCASS - 9th CEAS Conference, 10th EUCASS - 9th CEAS Conference ; Conference date: 09-07-2023 Through 13-07-2023. EUCASS, 2023. DOI: 10.13009/EUCASS2023-658.

[29] D. Krempus et al. *Organic Rankine Cycle Waste Heat Recovery System for Aircraft Auxiliary Power Units*. Towards Sustainable Aviation Summit. Toulouse, France, 2022.

[30] Dabo Krempus et al. “On mixtures as working fluids of air-cooled ORC bottoming power plants of gas turbines”. English. In: *Applied Thermal Engineering* 236 (2024). ISSN: 1359-4311. DOI: 10.1016/j.applthermaleng.2023.121730.

[31] J. J. Lee et al. “HISTORICAL AND FUTURE TRENDS IN AIRCRAFT PERFORMANCE, COST, AND EMISSIONS”. In: *Annual Review of Environment and Resources* 26.Volume 26, 2001 (2001), pp. 167–200. ISSN: 1545-2050. DOI: https://doi.org/10.1146/annurev.energy.26.1.167.

[32] S. Leijnen and F. van Veen. “The Neural Network Zoo”. In: *Proceedings* 47 (May 2020), p. 9. DOI: 10.3390/proceedings47010009.

[33] P. J. Linstrom and W. G. Mallard, eds. *NIST Chemistry WebBook*. NIST Standard Reference Database Number 69, National Institute of Standards and Technology, Gaithersburg MD, 20899, retrieved January 9, 2025. 2025. DOI: 10.18434/T4D303. URL: https://doi.org/10.18434/T4D303.

[34] M. Majer and M. Pini. “Design Guidelines for High-Pressure Ratio Supersonic Radial-Inflow Turbines of Organic Rankine Cycle Systems”. In: *Journal of the Global Power and Propulsion Society* (2024). In press.

[35] *Design Criteria and Efficiency Prediction for Radial Inflow Turbines*. Vol. Volume 1: Turbomachinery. Turbo Expo: Power for Land, Sea, and Air. May 1987, V001T01A086. DOI: 10.1115/87-GT-231. eprint: https://asmedigitalcollection.asme.org/GT/proceedings-pdf/GT1987/79238/V001T01A086/2397279/v001t01a086-87-gt-231.pdf. URL: https://doi.org/10.1115/87-GT-231.

[36] United Nations Environment Programme. *Paris Agreement*. UNTC XXVII 7.d. Dec. 12, 2015.

[37] B. Sarlioglu and C. T. Morris. “More Electric Aircraft: Review, Challenges, and Opportunities for Commercial Transport Aircraft”. In: *IEEE Transactions on Transportation Electrification* 1.1 (2015), pp. 54–64. DOI: 10.1109/TTE.2015.2426499.

[38] M. Schmidt and H. Lipson. “Symbolic Regression of Implicit Equations”. In: *Genetic Programming Theory and Practice VII*. Ed. by R. Riolo, U.-M. O’Reilly, and T. McConaghy. Boston, MA: Springer US, 2010, pp. 73–85. ISBN: 978-1-4419-1626-6. DOI: 10.1007/978-1-4419-1626-6_5.

[39] Delft University of Technology. *ORCHID – Organic Rankine Cycle Hybrid Integrated Device*. Accessed: 29-01-2025. URL: https://www.tudelft.nl/lr/organisatie/afdelingen/flow-physics-and-technology/flight-performance-and-propulsion/propulsion-power/facilities/orchid.

A

Evaluation Metrics

The surrogate model can be used to predict the efficiency and weight of an Organic Rankine Cycle (ORC) radial-inflow turbine for combined-cycle engines. The following chapter will explain the metrics that were used to evaluate the accuracy of the surrogate model. The results and precision of the surrogate models can be found in chapter 5.

There are many different metrics that can be used to indicate the accuracy of a model. This study looked at the Mean-Squared-Error (MSE) and R-squared (R^2). They will both be explained in section A.1 and section A.2, respectively.

A.1. Mean-squared-error

The Mean Squared Error (MSE) measures, as the name suggests, the average of the squared difference between predicted and actual values. A low MSE is preferred because it means that the difference between the actual and predicted values is small, resulting in a more accurate model. The MSE is sensitive to outliers in the dataset. This means that a model with good accuracy can have a high MSE when the data is wide spread. However, it can also be that the dataset has very little variation, resulting in a small MSE, but the predictions are not accurate.

The MSE is the loss function used by PySR to minimize the prediction error when training the surrogate model.

The MSE is calculated as follows:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{A.1}$$

where n is the number of data points, y_i the actual value generated by TurboSim and $\hat{y}$ the value generated by the surrogate model.

A.2. R-squared

The R-squared method indicates the proportion of variance in the dependent variable that the model explains. It ranges from 0 to 1, where 1 indicates a perfect fit. The R^2 metric can be used to compare models with different datasets, which is useful for this particular project because every fluid has its own surrogate model. However, it cannot give any information on how far the predictions are from the actual values. R^2 is not as affected by outliers as the MSE, because it looks at the overall trend in the data. This means that a model with a high MSE can still have a high R^2 and vice versa.

$$R^2 = 1 - \frac{SS_{res}}{SS_{tot}} \tag{A.2}$$

with

$$SS_{res} = \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \quad \text{(A.3)} \qquad SS_{tot} = \sum_{i=1}^{n}(y_i - \bar{y})^2 \quad \text{(A.4)}$$

where y_i are the actual values, $\hat{y}_i$ are the predicted values and $\bar{y}$ are the mean of the actual values.

B

Constraint violations

An overview of the constraint violations is presented per working fluid. The constraint violations for butane can be found in Table 5.3.

Table B.1: Constraint violations for working fluid cyclopentane expressed in percentage of the number of cases in the complete dataset

Constraint	VR min			VR mid			VR max		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1	-	-	-	-	-	-	-	-	-
2	5.28%	5.26%	5.20%	4.60%	4.67%	4.74%	3.67%	3.79%	3.81%
3	-	-	-	-	-	-	-	-	-
4	-	-	-	37.99%	38.00%	37.96%	43.15%	48.65%	48.69%
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	33.29%	33.31%	33.35%	33.31%	33.31%	33.29%	33.31%	33.34%	33.33%
8	20.73%	20.78%	20.71%	26.73%	26.74%	26.70%	26.66%	26.52%	26.55%
9	100.00%	97.65%	91.77%	98.07%	64.61%	51.39%	91.07%	48.18%	38.45%
10	0.22%	-	-	3.99%	-	-	5.80%	-	-
11	67.14%	1.23%	0.02%	70.03%	6.94%	1.13%	69.85%	9.55%	3.16%
12	66.13%	25.45%	11.68%	34.93%	6.92%	5.61%	18.52%	6.24%	6.29%
13	21.02%	-	-	-	-	-	-	-	-

Table B.2: Constraint violations for working fluid ethanol expressed in percentage of the number of cases in the complete dataset

Constraint	VR min			VR mid			VR max		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1	-	-	-	-	-	-	7.12%	7.09	7.15
2	4.97%	4.93%	4.92%	5.65%	5.90%	5.88%	7.12%	7.09%	7.15%
3	-	-	-	-	-	-	-	-	-
4	7.36	7.36	7.36	66.34%	66.41%	66.37%	73.12%	73.09%	73.12%
5	-	-	-	25.10	25.09	25.08	41.90	41.99	42.04
6	-	-	-	-	-	-	-	-	-
7	33.36%	33.40%	33.38%	33.29%	33.33%	33.33%	33.33%	33.43%	33.24%
8	28.15%	28.15%	28.14%	18.49%	18.52%	18.48%	13.16%	13.12%	13.06%
9	100.00%	95.35%	86.36%	58.04%	25.58%	16.86%	41.89%	11.09%	2.50%
10	0.99%	-	-	19.08%	-	-	21.76%	-	-
11	72.30%	3.80%	0.19%	79.37%	26.80%	14.06%	78.73%	28.73%	15.80%
12	61.33%	19.64%	7.80%	6.71%	6.03%	5.87%	4.62%	3.49%	3.45%
13	13.51%	-	-	-	-	-	-	-	-

Table B.3: Constraint violations for working fluid MM expressed in percentage of the number of cases in the complete dataset

Constraint	VR min			VR mid			VR max		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1	-	-	-	-	-	-	-	-	-
2	4.67%	4.73%	4.68%	4.13%	4.14%	4.19%	4.31%	4.34%	4.34%
3	-	-	-	-	-	-	-	-	-
4	1.03	1.03	1.04	45.38%	45.37%	45.34%	53.55%	53.49%	53.53%
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	33.31%	33.33%	33.34%	33.26%	33.37%	33.28%	33.27%	33.29%	33.29%
8	25.00%	25.00%	24.97%	27.16%	27.16%	27.22%	33.27%	33.29%	33.29%
9	99.99%	90.02%	78.32%	80.07%	40.40%	31.40%	66.69%	30.27%	20.95%
10	-	-	-	0.22%	-	-	1.17%	-	-
11	43.29%	0.05%	-	47.68%	2.47%	-	47.44%	3.87%	0.03%
12	50.82%	10.21%	6.66%	12.17%	5.43%	4.51%	6.55%	4.55%	4.35%
13	3.87%	-	-	-	-	-	-	-	-

Table B.4: Constraint violations for working fluid R134a expressed in percentage of the number of cases in the complete dataset

Constraint	VR min			VR mid			VR max		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1	-	-	-	-	-	-	-	-	-
2	5.21%	5.30%	5.29%	5.28%	5.33%	5.33%	4.70%	4.73%	4.72%
3	-	-	-	-	-	-	-	-	-
4	-	-	-	-	-	-	2.98%	2.98%	2.98%
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	33.32%	33.30%	33.33%	33.35%	33.31%	33.33%	33.29%	33.31%	33.32%
8	18.08%	18.13%	18.12%	21.37%	21.43%	21.41%	25.882%	25.81%	25.82%
9	100.00%	99.52%	96.53%	100.00%	98.87%	94.85%	100.00%	97.89%	92.22%
10	-	-	-	0.76%	-	-	1.82%	-	-
11	67.96%	0.09	-	77.44%	2.85%	0.19%	78.75%	4.90%	0.45
12	72.32%	28.30%	18.42%	70.81%	31.03%	17.67%	67.50%	28.37%	13.12%
13	35.35%	-	-	30.50%	-	-	25.32%	-	-

Table B.5: Constraint violations for working fluid toluene expressed in percentage of the number of cases in the complete dataset

Constraint	VR min			VR mid			VR max		
	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$	$\dot{m} = 0.5$	$\dot{m} = 3.0$	$\dot{m} = 5.0$
1	-	-	-	-	-	-	2.50%	2.50%	2.50%
2	5.41%	5.33%	5.34%	4.24%	4.35%	4.39%	3.40%	3.44%	3.45%
3	-	-	-	-	-	-	-	-	-
4	20.48%	20.49%	20.47%	65.18%	65.32%	65.35%	70.33%	70.37%	70.43%
5	-	-	-	-	-	-	-	-	-
6	-	-	-	-	-	-	-	-	-
7	33.29%	33.41%	33.35%	33.27%	33.31%	33.32%	33.33%	33.35%	33.33%
8	28.39%	28.42%	28.42%	22.55%	22.63%	22.68%	19.43%	19.45%	19.42%
9	99.96%	83.47%	71.15%	56.99%	24.11%	14.21%	42.45%	10.95%	3.70%
10	0.93%	-	-	9.29%	-	-	14.92%	-	-
11	66.32%	3.76%	0.14%	64.24%	16.68%	6.81%	67.04%	20.77%	8.94%
12	45.99%	7.75%	6.55%	7.98%	2.40%	2.41%	6.27%	4.14%	3.62%
13	1.50%	-	-	-	-	-	-	-	-

C

Dataset distribution

The dataset distributions of the six working fluids are presented below.

C.1. Design variables

The distribution of the design variables, as well as the mass flow rate and volumetric flow ratio are presented below.

C.1.1. Complete dataset

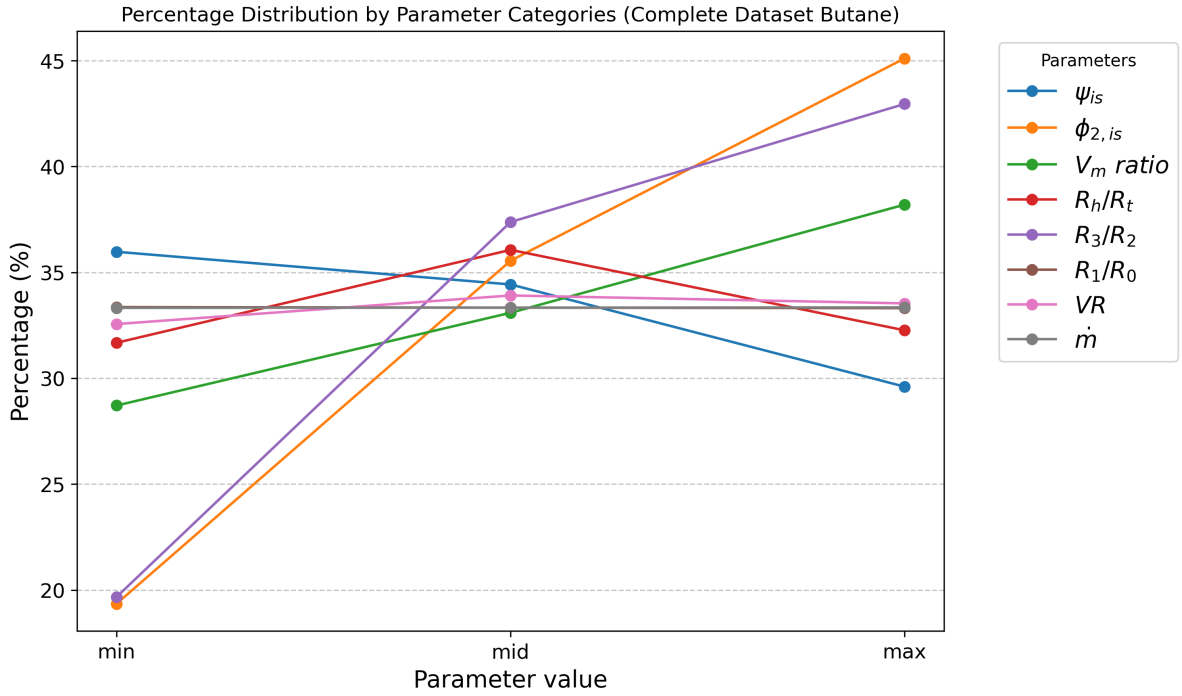


Figure C.1: Distribution of parameters in complete dataset for butane

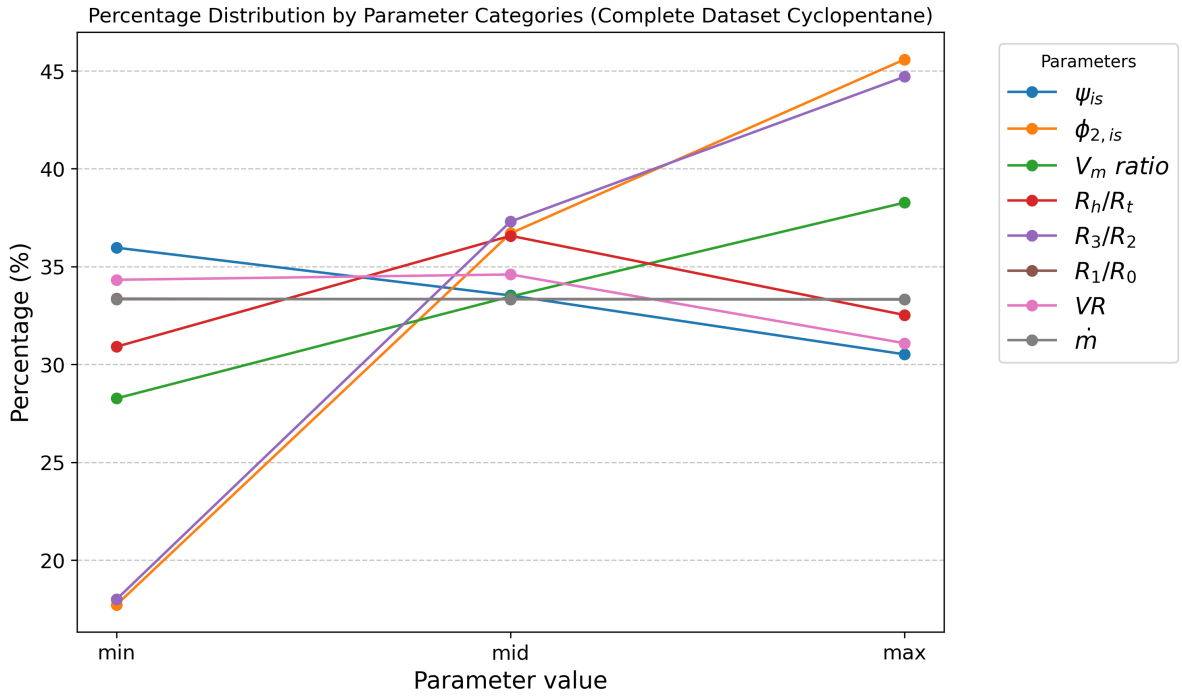


Figure C.2: Distribution of parameters in complete dataset for cyclopentane

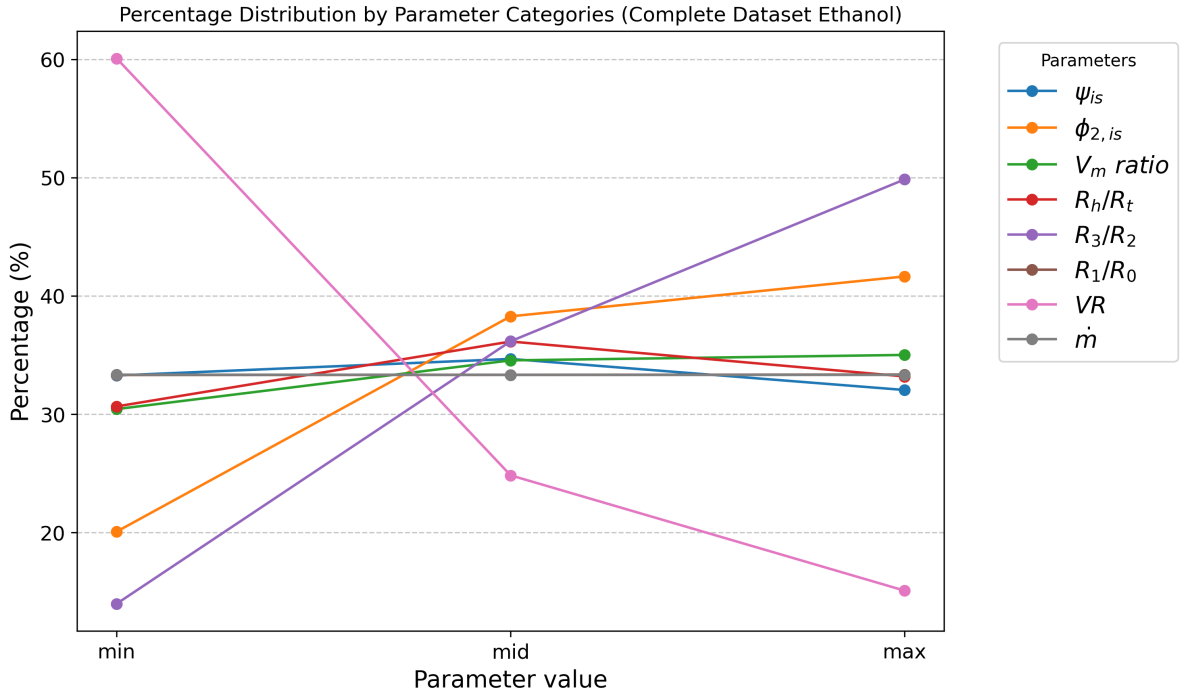


Figure C.3: Distribution of parameters in complete dataset for ethanol

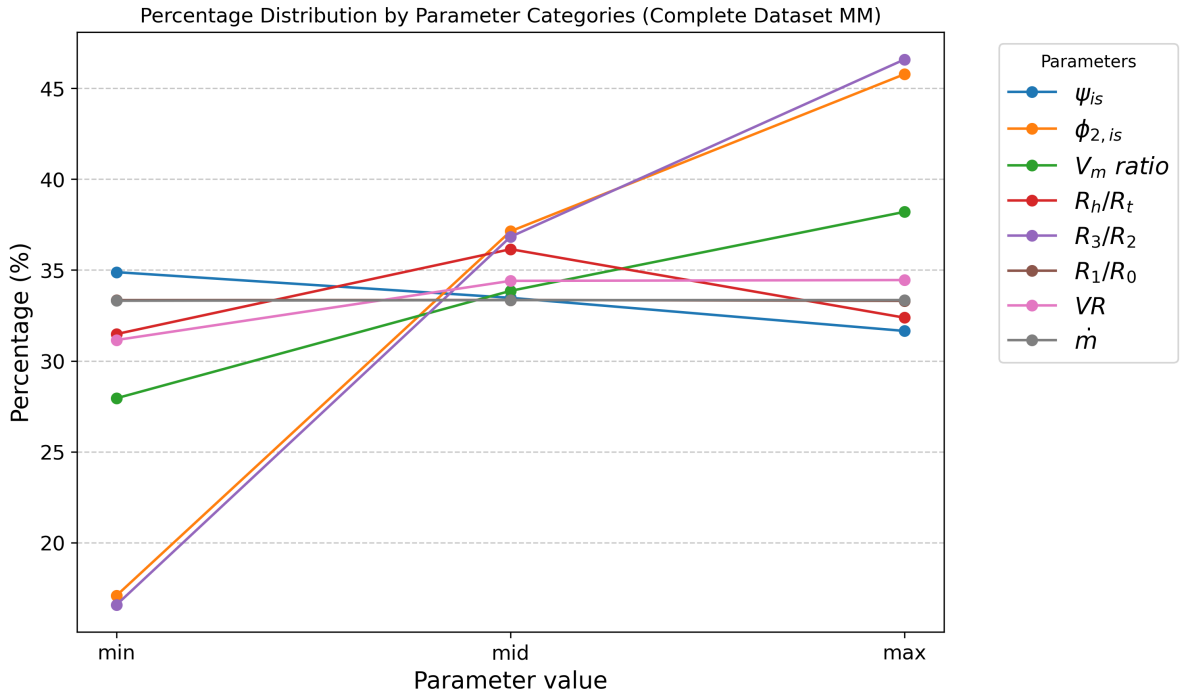


Figure C.4: Distribution of how the training dataset is made up for working fluid MM

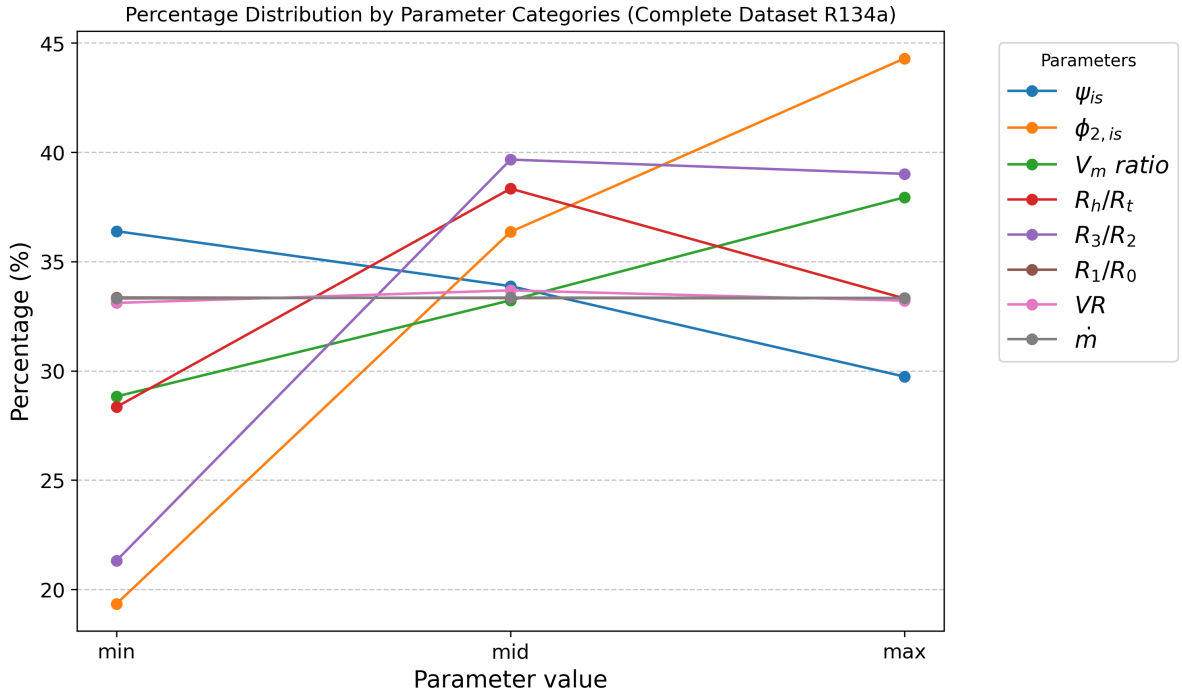


Figure C.5: Distribution of parameters in complete dataset for R134a

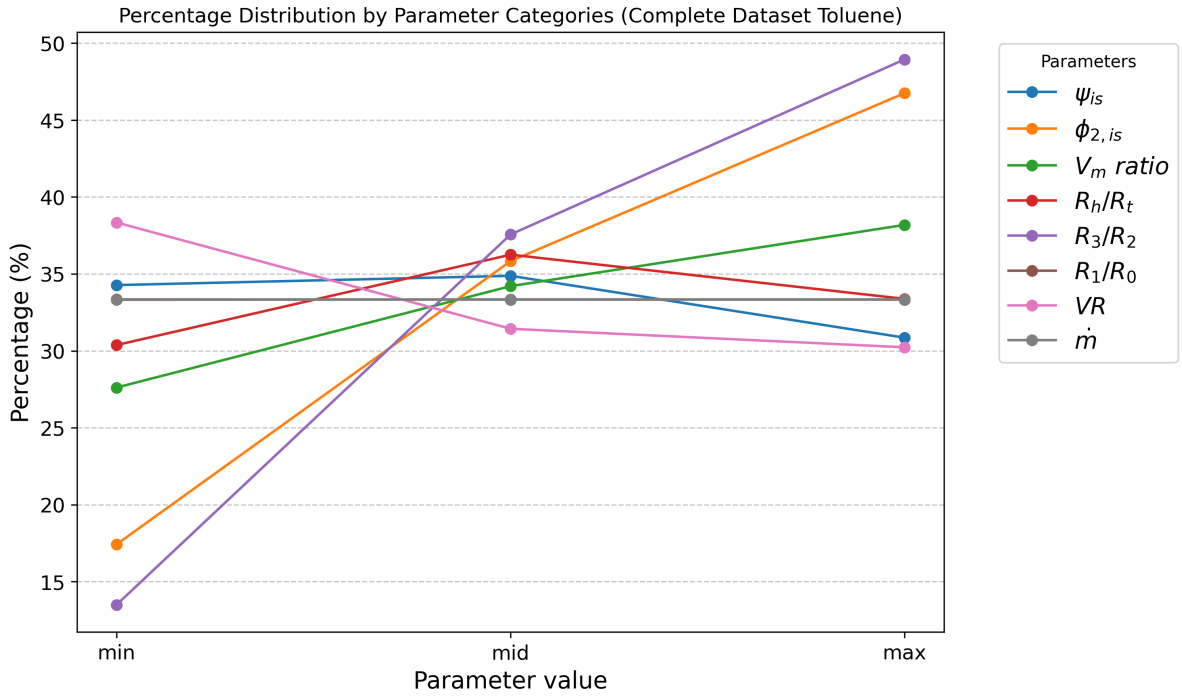


Figure C.6: Distribution of parameters in complete dataset for toluene

C.1.2. Reduced dataset

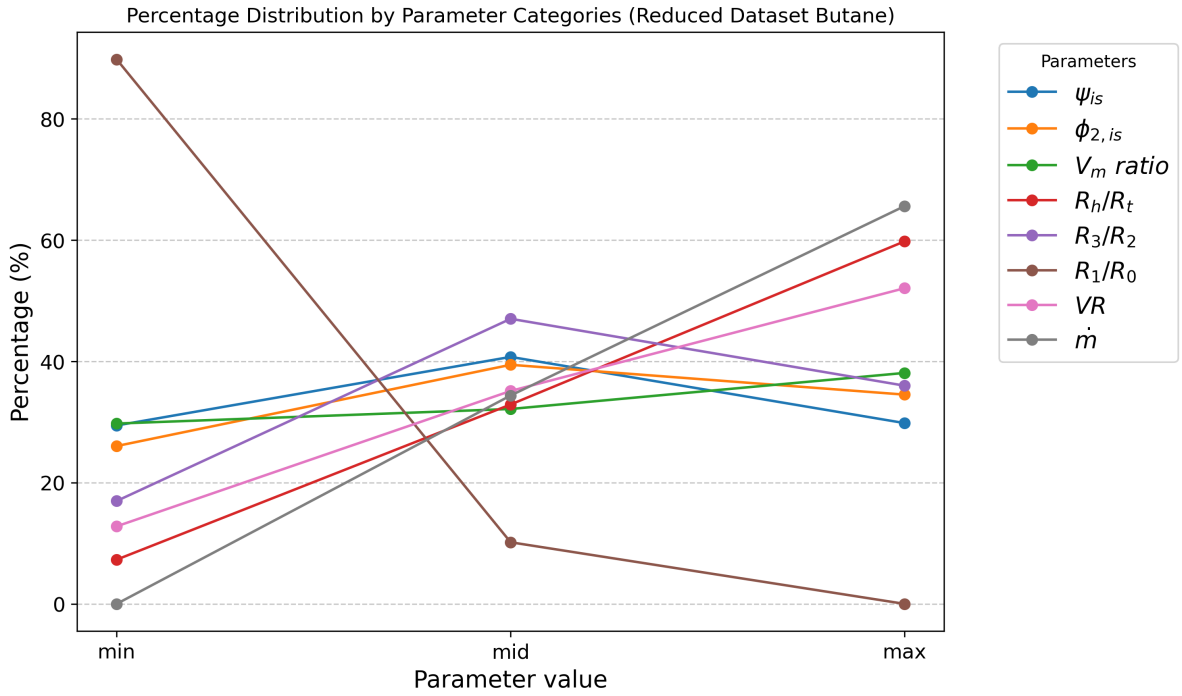


Figure C.7: Distribution of parameters in reduced dataset for butane

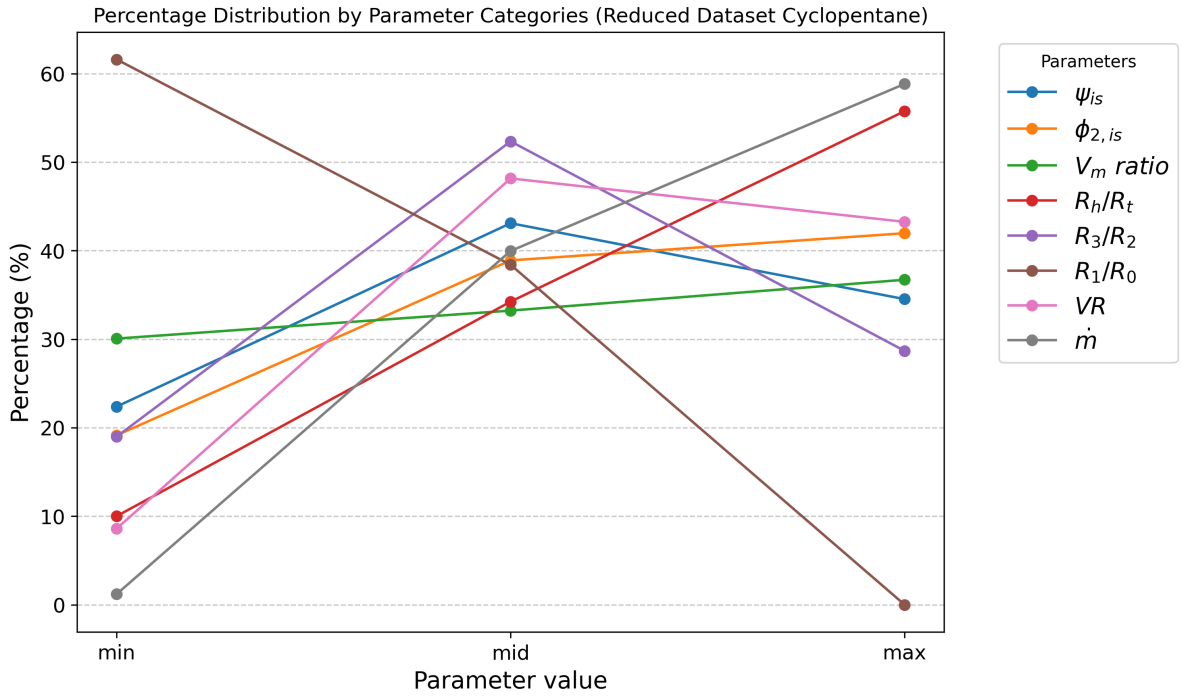


Figure C.8: Distribution of parameters in reduced dataset for cyclopentane

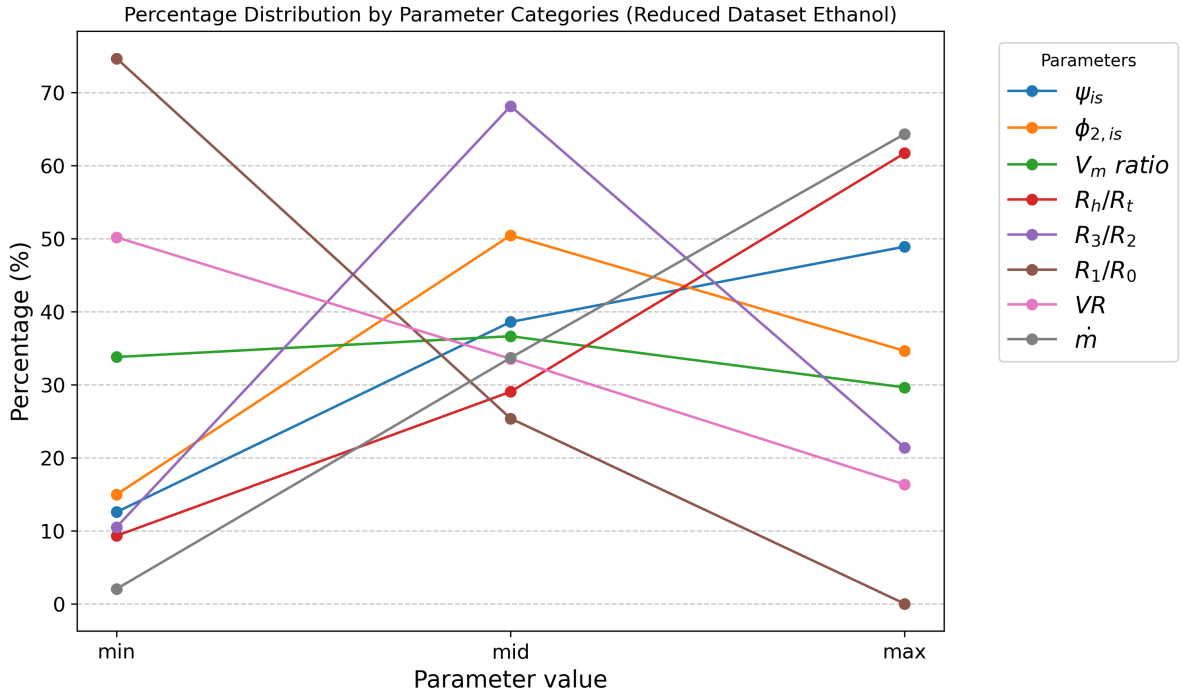


Figure C.9: Distribution of parameters in reduced dataset for ethanol

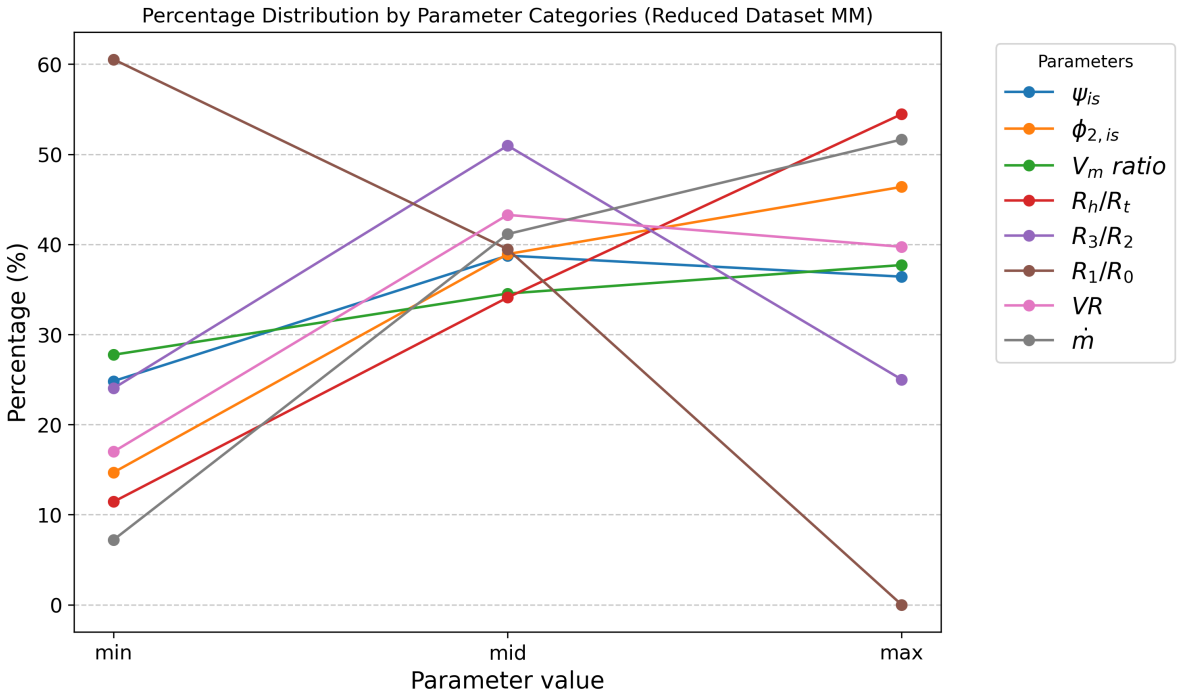


Figure C.10: Distribution of how the reduced training dataset is made up for working fluid MM

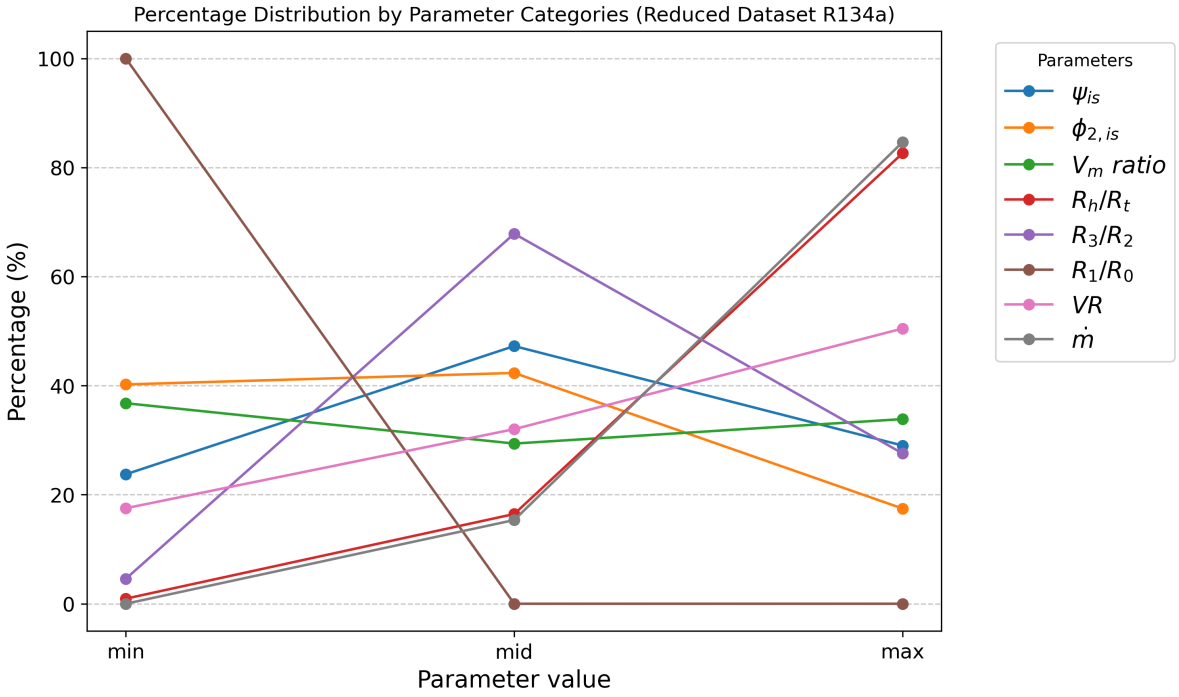


Figure C.11: Distribution of parameters in reduced dataset for R134a

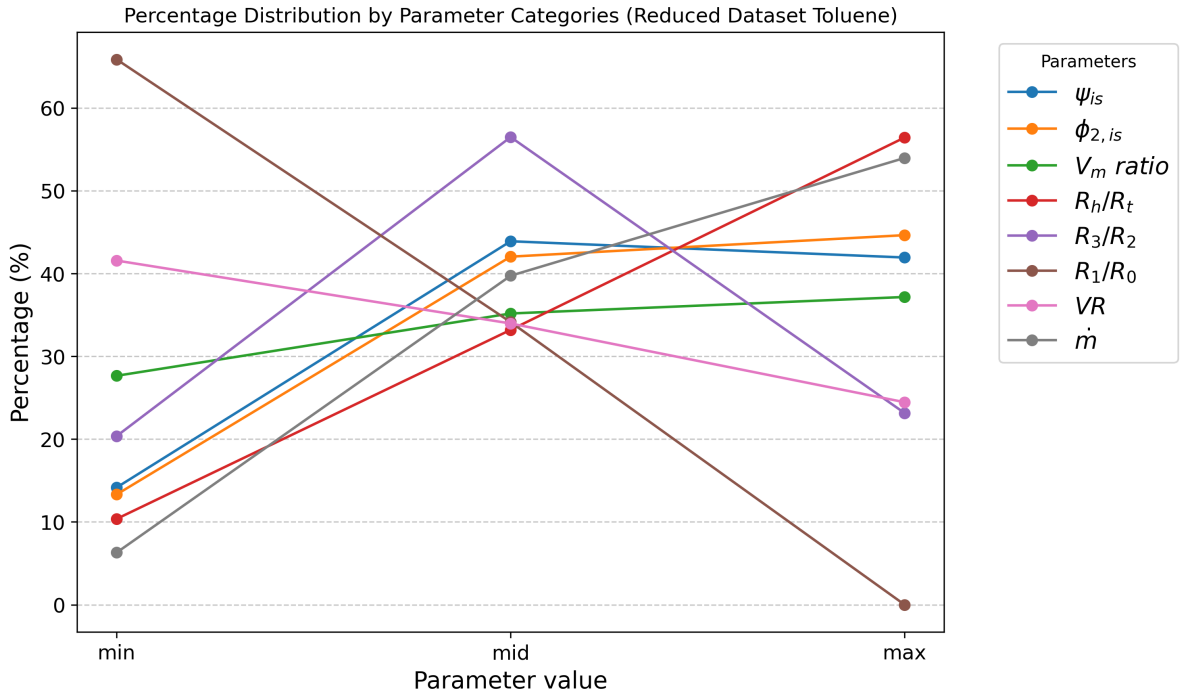


Figure C.12: Distribution of parameters in reduced dataset for toluene

C.2. Reduced temperature

Figure C.13 to Figure C.18 display the number of cases per investigated T_r. It can be seen that most of the cases operate at higher reduced temperatures.

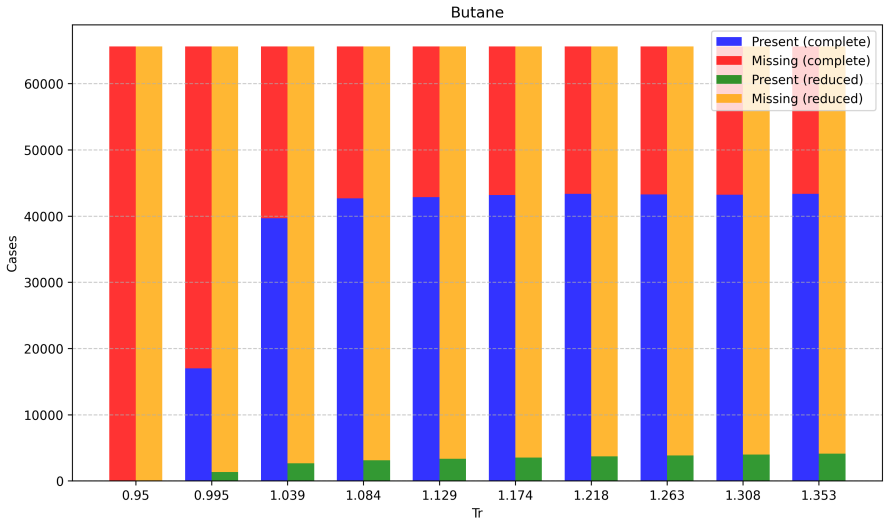


Figure C.13: Number of cases generated for input parameter reduced temperature, working fluid butane

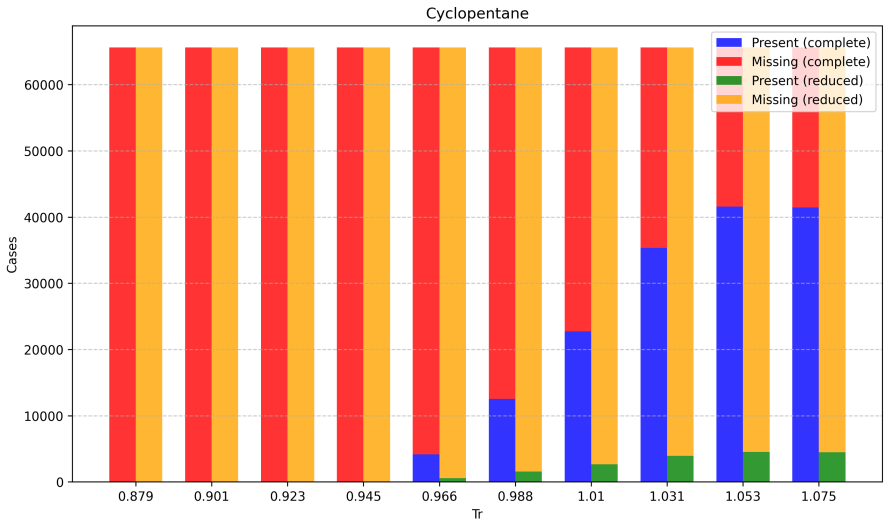


Figure C.14: Number of cases generated for input parameter reduced temperature, working fluid cyclopentane

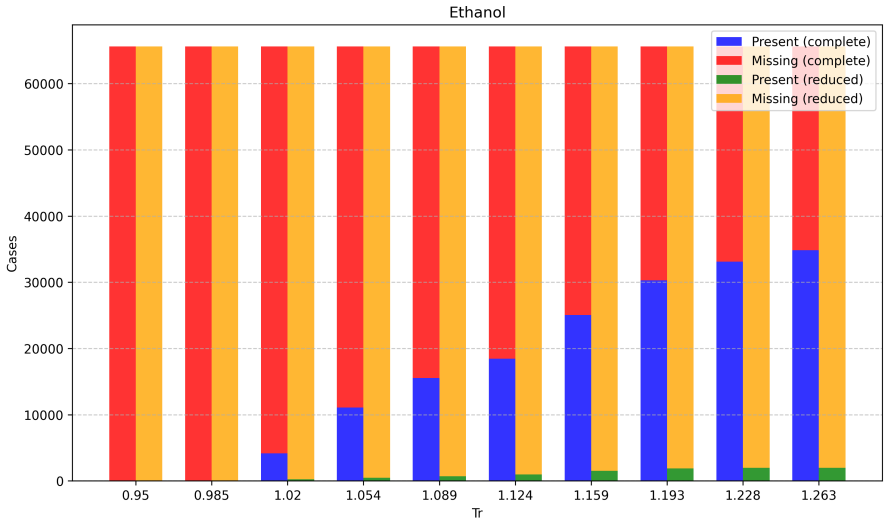


Figure C.15: Number of cases generated for input parameter reduced temperature, working fluid ethanol

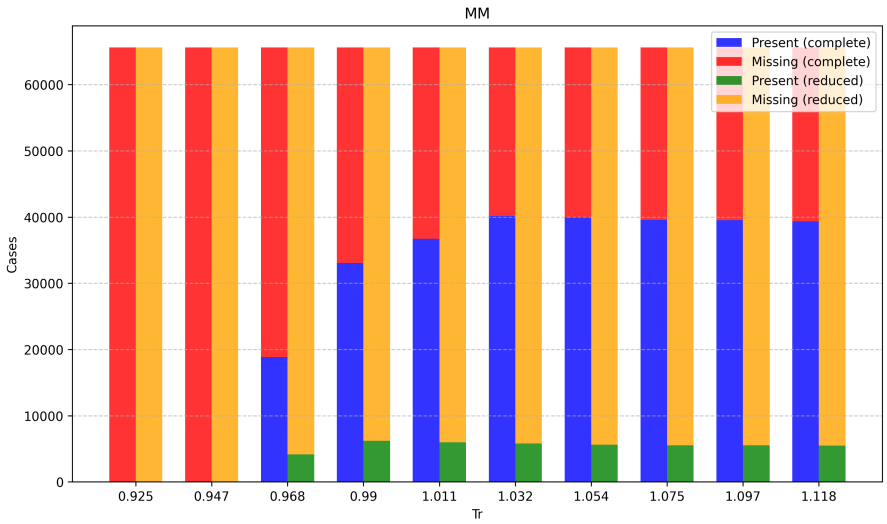


Figure C.16: Number of cases generated for input parameter reduced temperature, working fluid MM

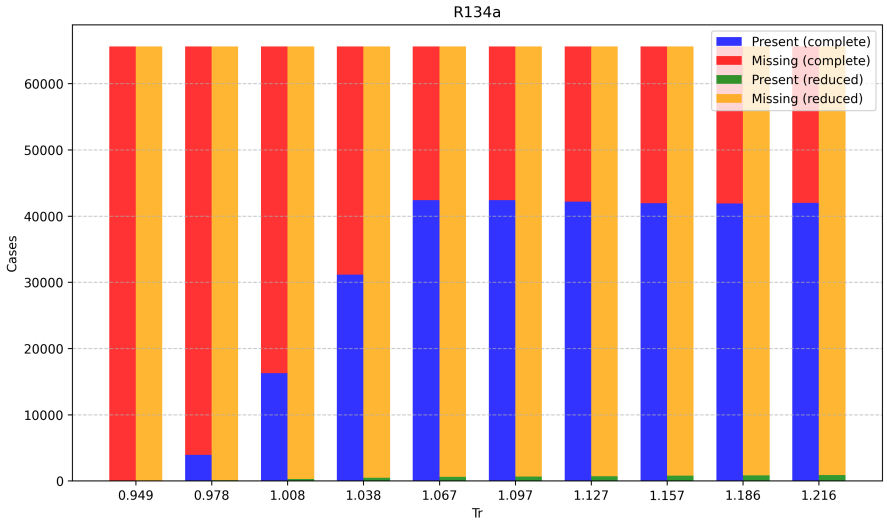


Figure C.17: Number of cases generated for input parameter reduced temperature, working fluid R134a

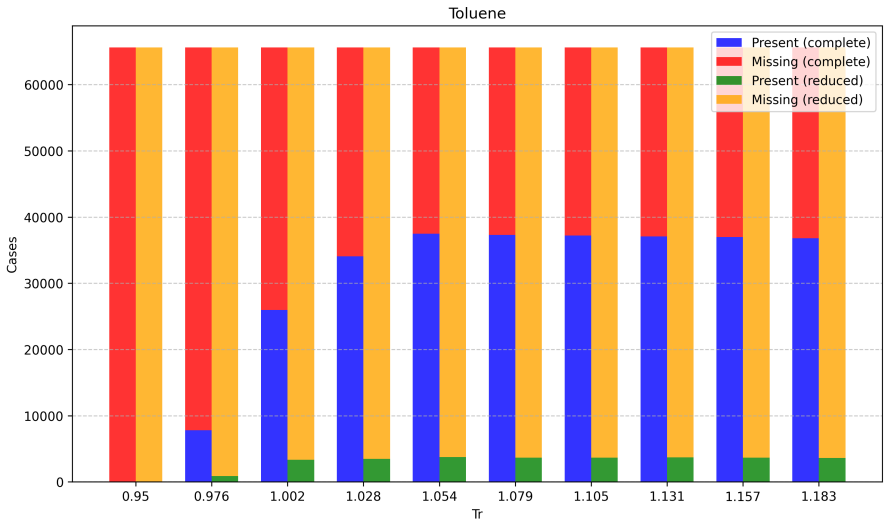


Figure C.18: Number of cases generated for input parameter reduced temperature, working fluid toluene

D

Analysis of R^2 Consistency for Training and Test Data

A 90/10% split of the data was selected oven an 80/20% split due to the large size of the dataset. A check is performed to see if this data split is correct. The R^2 value will be calculated for two datasets: 1. it will use 10% of the dataset that was set aside during training ('new data'), 2. the same number of cases is randomly selected from the dataset using in training ('seen data').

For every working fluid, the best performing R^2 value and the average R^2 value obtained during training are presented in Table D.1 to D.6, with the highest value highlighted in blue. The number of tested cases varies across fluids due to the 24-hour runtime limit on the Delft Blue Supercomputer. Since the training time depends significantly on the selected PySR settings, some runs did not complete within the allowed time and had to be repeated. Each fluid was tested with 188 unique training settings unless explicitly stated otherwise.

The difference between the average R^2 values is small, indicating that a 90/10% data split provides accurate results for both training and unseen data. As expected, the R^2 values of the seen data are generally slightly higher than those for new data, as the model was directly trained on them.

Table D.1: Average and best R^2 scores for butane from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.836965	0.837272	0.00031	0.88655	0.889702	0.00315
		60	0.856255	0.856910	0.00065	0.932592	0.92987	0.00272
	Complete	40	0.844828	0.844672	0.000155	0.909871	0.90780	0.00198
		60 (205 cases)	0.851922	0.851895	0.000027	0.91761	0.917911	0.00030
Weight	Reduced	40	0.557602	0.55875	0.00114	0.70809	0.710905	0.00282
		60	0.596634	0.596096	0.000537	0.705616	0.68578	0.01984
	Complete	40 (194 cases)	0.761727	0.761797	0.00007	0.828104	0.82804	0.00006
		60 (220 cases)	0.781845	0.781511	0.000334	0.856571	0.85565	0.00092

Table D.2: Average and best R^2 scores for cyclopentane from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.825023	0.825409	0.000386	0.888877	0.88625	0.00262
		60	0.853573	0.852989	0.000584	0.914051	0.90604	0.00801
	Complete	40	0.839888	0.840048	0.00016	0.909318	0.90825	0.00107
		60	0.849383	0.849467	0.00008	0.91044	0.912017	0.00158
Weight	Reduced	40	0.707925	0.707083	0.000842	0.807935	0.78871	0.01922
		60	0.752112	0.747523	0.004589	0.83623	0.841421	0.00519
	Complete	40 (192 cases)	0.811208	0.811259	0.000051	0.877597	0.86974	0.00785
		60 (193 cases)	0.844862	0.844196	0.000666	0.892234	0.88704	0.00520

Table D.3: Average and best R^2 scores for ethanol from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.819767	0.816496	0.003271	0.87905	0.886521	0.00747
		60	0.842998	0.843126	0.000128	0.900271	0.88841	0.01186
	Complete	40	0.839765	0.839677	0.000088	0.89941	0.902885	0.00348
		60	0.847515	0.84745	0.000065	0.90571	0.908412	0.00270
Weight	Reduced	40	0.890137	0.889601	0.000536	0.91817	0.92756	0.00939
		60	0.900767	0.899981	0.000786	0.938202	0.92916	0.00904
	Complete	40 (192 cases)	0.777327	0.77802	0.000693	0.886825	0.88533	0.00150
		60 (192 cases)	0.809943	0.811079	0.001136	0.891542	0.87884	0.01270

Table D.4: Average and best R^2 scores for MM from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.848351	0.847689	0.000662	0.90751	0.912831	0.00532
		60	0.873696	0.874118	0.000422	0.93563	0.936593	0.00096
	Complete	40	0.842165	0.84213	0.000035	0.905286	0.904900	0.00039
		60	0.850504	0.85016	0.000344	0.93788	0.938008	0.00013
Weight	Reduced	40	0.791927	0.791832	0.000095	0.8673278	0.86772	0.00061
		60	0.815357	0.815446	0.000089	0.884995	0.87879	0.00620
	Complete	40 (195 cases)	0.780128	0.780664	0.000536	0.85337	0.885769	0.03240
		60 (229 cases)	0.801378	0.801147	0.000231	0.877003	0.87398	0.00302

Table D.5: Average and best R^2 scores for R134a from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.848964	0.84631	0.002654	0.90061	0.919854	0.01925
		60	0.866122	865167	0.000955	0.915559	0.88641	0.02915
	Complete	40	0.847309	0.847406	0.000097	0.91135	0.913421	0.00207
		60	0.856653	0.856801	0.000148	0.92341	0.92147	0.00194
Weight	Reduced	40	0.265624	0.264983	0.000641	0.51041	0.51346	0.00305
		60	0.307856	0.306972	0.000884	0.568717	0.51255	0.05617
	Complete	40 (195 cases)	0.752491	0.752339	0.000152	0.842891	0.83585	0.00704
		60 (229 cases)	0.782343	0.782015	0.000328	0.88693	0.891962	0.00503

Table D.6: Average and best R^2 scores for toluene from training phase

Trained for	Type dataset	Maxsize	Average R^2			Best R^2		
			Seen data	New data	|Difference|	Seen data	New data	|Difference|
Efficiency	Reduced	40	0.837129	0.836791	0.000338	0.905298	0.90495	0.00035
		60	0.860099	860137	0.000038	0.90912	0.90949	0.00037
	Complete	40	0.834621	0.834424	0.000197	0.895204	0.89422	0.00099
		60	0.850221	0.850011	0.000210	0.91005	0.910365	0.00194
Weight	Reduced	40	0.891984	0.891817	0.000167	0.91747	0.926218	0.00875
		60 (197 cases)	0.897937	0.897509	0.000428	0.931205	0.92258	0.00862
	Complete	40 (192 cases)	0.793623	0.79092	0.002703	0.866797	0.85917	0.00762
		60 (230 cases)	0.821907	0.822283	0.000376	0.90056	0.905253	0.00470

E

Surrogate expressions

The equations corresponding to the highest scoring R^2 models trained on the complete dataset are presented below.They are grouped by working fluid The differences and similarities in the surrogate models and symbolic expressions will be discussed. The training settings for each fluid are tabulated and the operators that appear in the expressions are indicated in blue. An overview of all the parameters is also included in each section. This can give an insight on which settings are preferred during training.

E.1. Ethanol

$$\eta_{tt_{net_{Eth}}} = 34.4 \cdot R_3/R_2 - 2 \cdot Z - \ln(VR) + 86.1 + \frac{34.4 \cdot (-R_h/R_t + 1.60 \cdot \tan(R_3/R_2) - 0.972)}{\psi_{is} \cdot (3.56 - 4.19 \cdot \phi_{2,is})} - \frac{2.60}{\phi_{2,is}} \tag{E.1}$$

$$W_{Eth} = \left| \frac{R_h/R_t \cdot VR \cdot \dot{m} \cdot \sin(Z - \tan(R_3/R_2))}{R_1/R_0 \cdot Vm_{ratio}\,(\phi_{2,is} \cdot 4.37 - \psi_{is} \cdot \tan(R_h/R_t))} \right| \tag{E.2}$$

E.2. Refrigerant R134a

$$\eta_{tt_{net_{R134a}}} = -VR + \frac{24.4 + \frac{(6.02 \cdot \phi_{2,is} - 0.0747)\sin\left(\psi_{is} \cdot (-0.542) + R_h/R_t \cdot 1.99 + \frac{1.59}{R_3/R_2}\right)}{\psi_{is} \cdot \frac{1}{Vm_{ratio}}}}{0.268} + \tan(0.652) \tag{E.3}$$

$$= 0.011 - VR + \frac{24.4 + \frac{(-(-1) \cdot 6.02 \cdot \phi_{2,is} - 0.0747)\sin\left(\psi_{is} \cdot (-0.542) + R_h/R_t \cdot 1.99 + \frac{1.59}{R_3/R_2}\right)}{\psi_{is} \cdot \frac{1}{Vm_{ratio}}}}{0.268} \tag{E.4}$$

$$W_{R134a} = \left| \frac{Z \cdot \dot{m} \cdot \tan\left(\frac{R_h/R_t}{R_1/R_0 + \frac{\left| \frac{1.47 \cdot \phi_{2,is} \cdot Vm_{ratio}^2 \cdot (\cos(R_h/R_t) - \tan(\tan(\tan(\cos(R_3/R_2)))))}{\psi_{is}} - 2.04 \cdot \tan\left(\tan\left(\tan\left(R_h/R_t - 0.0697\right)\right)\right) \right|}{VR}} \right)}{\phi_{2,is} + R_1/R_0} \right| \tag{E.5}$$

E.3. Butane

$$\eta_{tt_{net_{But}}} = 6.43 \cdot \psi_{is} - \ln(VR) + 84.1 + \frac{6.43 \cdot Vm_{ratio} \cdot \sin\left(1.89 \cdot \cos\left(\frac{\cos(\cos(1.32 \cdot R_h/R_t))}{\sin(R_3/R_2) - 0.103}\right)\right)}{\psi_{is}\,(-\ln(\phi_{2,is}) - 0.0558)} \tag{E.6}$$

$$W_{But} = 0.185 \left| \frac{\psi_{is} \cdot Z \cdot \dot{m} \cdot (VR - \ln(0.141 \cdot \psi_{is})) \cdot \left(\left| \frac{R_h/R_t + 1.03}{R_3/R_2} \right| - 2.16 \right) \cdot \left| \frac{2.72 - \frac{0.885}{R_3/R_2}}{\phi_{2,is}} \right|}{R_1/R_0 \cdot Vm_{ratio} \cdot (\psi_{is} + Vm_{ratio})} \right| \tag{E.7}$$

E.4. Cyclopentane

$$\eta_{tt_{net_{Cyclo}}} = \frac{Vm_{ratio} \left[\left(-R_h/R_t + \sin\left(\sin\left(\sin\left(\frac{VR}{(-1) \cdot 0.419 \frac{1}{R_3/R_2}} \right) + 0.117 - \frac{R_h/R_t}{\psi_{is}} \right) \right) 1.73 \right) \phi_{2,is} \cdot 12.4 + \sin(VR) \right]}{\psi_{is}} + \psi_{is} - 0.265 + 86.8 \tag{E.8}$$

$$= \frac{Vm_{ratio} \left[\left(-R_h/R_t + \sin\left(\sin\left(\sin\left(\frac{VR}{-0.419 \frac{1}{R_3/R_2}} \right) + 0.117 - \frac{R_h/R_t}{\psi_{is}} \right) \right) 1.73 \right) \phi_{2,is} \cdot 12.4 + \sin(VR) \right]}{\psi_{is}} + 86.535 + \psi_{is} \tag{E.9}$$

$$W_{Cyclo} = \left| VR \cdot \left(0.0113 + \frac{\dot{m} \cdot \sin(-0.764) \cdot \sin\left(-\frac{0.0609}{R_3/R_2 - 0.407} \right)}{Vm_{ratio} \cdot R_1/R_0 \cdot \left(\frac{\phi_{2,is}}{0.487 \cdot (-0.544) \cdot R_h/R_t \cdot Z} + \psi_{is} \right)} \right) \right| \tag{E.10}$$

E.5. Toluene

$$\eta_{tt_{net_{Tol}}} = \phi_{2,is} - \tan(VR \cdot \tan(VR)) + 81.8 + \frac{16.8 \cdot \sin[\sin(0.571 \cdot \psi_{is} + R_3/R_2 - \tan[R_h/R_t \cdot \sin(R_3/R_2) - \ln(R_3/R_2)])]}{\psi_{is} - Vm_{ratio} \cdot (\phi_{2,is} - 0.396)} \tag{E.11}$$

$$W_{Tol} = |x| - 0.938 \tag{E.12}$$

with

$$x = (2 \cdot \phi_{2,is} - \dot{m}) \cdot \cos(R_1/R_0) \cdot \tan(R_h/R_t) \cdot \tan(\tan(Z)) \cdot \left| \frac{\cos(R_3/R_2 - 1.74) \cdot |VR|}{\left| Vm_{ratio} \cdot \left(\psi_{is} + \cos(\psi_{is}) + \cos\left(\frac{20.6}{\cos(R_3/R_2)} - \frac{\psi_{is} R_h/R_t}{\phi_{2,is}} \right) \right) \right|} \right| \tag{E.13}$$

E.6. Siloxane MM

$$\eta_{tt_{net_{MM}}} = \phi_{2,is} + R_1/R_0 + \frac{Vm_{ratio} \sin\left(\frac{R_h/R_t + 1.01 + \frac{0.328}{\psi_{is}}}{R_3/R_2} \right)}{0.212 \cdot 0.219 \cdot \frac{1}{\phi_{2,is}} \cdot \psi_{is}} - \frac{\cos\left(\frac{VR}{1.05}\right)}{0.356} - 1 \cdot -1.55 + 83.8 + \frac{0.472}{Z} \tag{E.14}$$

$$= 85.35 + \frac{0.472}{Z} + \phi_{2,is} + R_1/R_0 + \frac{Vm_{ratio} \sin\left(\frac{R_h/R_t + 1.01 + \frac{0.328}{\psi_{is}}}{R_3/R_2} \right)}{0.212 \cdot 0.219 \cdot \frac{1}{\phi_{2,is}} \cdot \psi_{is}} - \frac{\cos\left(\frac{VR}{1.05}\right)}{0.356} \tag{E.15}$$

$$W_{MM} = \left| \frac{\psi_{is} \cdot R_h/R_t^2 \cdot \left(R_3/R_2 - \frac{\dot{m}}{\tan(\phi_{2,is})} \right) \cdot \left(\frac{VR}{4.46 \cdot Vm_{ratio}} - 0.707 \right) \cdot \tan(\tan(Z))}{R_1/R_0 \cdot \tan(\tan(R_3/R_2 + 1.21))} \right| \tag{E.16}$$

E.7. Training settings for surrogate models trained on the complete dataset

The selected operators in the training process are tabulated in Table E.1. The nesting constraint settings are also indicated. Self nesting was allowed up to a depth of 1, meaning that $\sin(\sin(x))$ is allowed. When nothing was specified, PySR has the freedom to nest the operators indefinitely,which clearly results in deep nesting, as can be seen in Equation E.6 and E.8.

The binary operator *power* (^) was never included in the best performing model expressions, while *sin* and *tan* are always included when they were specified. The type of constraints specified did not affect the complexity of the best performing surrogate models, because they all have a complexity between 31 and 40, when a maximum complexity of 60 was allowed.

Table E.1: Used operators in efficiency surrogate equations using complete dataset. The operators indicated in blue are appearing in the final equation

Working fluid	**Binary operator**	**Unary operators**					**Type of nesting allowed**		
	Power (^)	Sin	Cos	Tan	Ln	Exp	No nesting	Self nesting	Nothing specified
Ethanol	✓			✓	✓		✓		
R134a		✓	✓	✓			✓		
Butane		✓	✓		✓				✓
Cyclopentane		✓	✓		✓				✓
Toluene	✓	✓		✓	✓				✓
MM	✓	✓	✓					✓	

Similarly to the efficiency expressions trained on the complete dataset, operators *sin* and *tan* are always included when they were specified during training. The PySR manual said to specify the least amount of operators to choose from during training. In the case of trigonometric functions *sin* and *cos*, it is suggest to only specify one, because the other one can be found by adding a phase shift. One can clearly see that when both were used during training, only *sin* was included in the expression, making *cos* redundant. Thus, it is recommended to only include one of them in further studies using PySR as the training model.

Table E.2: Used operators in weight surrogate equations using complete dataset. The operators indicated in blue are appearing in the final equation. Note the absolute value was always specified.

Working fluid	**Binary operator**	**Unary operators**					**Type of nesting allowed**		
	Power	Sin	Cos	Tan	Ln	Exp	No nesting	Self nesting	Nothing specified
Ethanol		✓	✓	✓					✓
R134a	✓		✓	✓		✓			✓
Butane					✓	✓	✓		
Cyclopentane		✓	✓					✓	
Toluene	✓		✓	✓		✓		✓	
MM			✓	✓					✓

E.8. Overview of operators used in expressions

Table E.3 and E.4 present the design variables and input parameters used in the expressions above. The expressions for turbine weight include most of these parameters, whereas the efficiency expressions primarily include those with the highest influence of efficiency. This distinction arises because the impact of these parameters on weight is generally more significant than on efficiency, as discussed in section 4.2.

Table E.3: Design variables and input parameters included in surrogate equations for efficiency predictions based on the complete dataset

Working fluid	Binary operator	Unary operators					Type of nesting allowed		
	$\phi_{2,is}$	ψ_{is}	R_1/R_0	R_3/R_2	R_h/R_t	V_m ratio	Z	$\dot{m}$	VR
Ethanol	✓	✓		✓	✓		✓		✓
R134a	✓	✓		✓	✓	✓			✓
Butane	✓	✓		✓	✓	✓			✓
Cyclopentane	✓	✓		✓	✓	✓			✓
Toluene	✓	✓		✓	✓	✓			✓
MM	✓	✓	✓	✓	✓	✓	✓		✓

Table E.4: Design variables and input parameters included in surrogate equations for weight predictions based on the complete dataset

Working fluid	Binary operator	Unary operators					Type of nesting allowed		
	$\phi_{2,is}$	ψ_{is}	R_1/R_0	R_3/R_2	R_h/R_t	V_m ratio	Z	$\dot{m}$	VR
Ethanol	✓	✓	✓	✓	✓	✓	✓	✓	✓
R134a	✓	✓	✓	✓	✓	✓	✓	✓	✓
Butane	✓	✓		✓	✓	✓	✓	✓	✓
Cyclopentane	✓	✓	✓	✓	✓	✓		✓	✓
Toluene	✓	✓	✓	✓	✓	✓	✓	✓	✓
MM	✓	✓	✓	✓	✓	✓	✓	✓	✓

E.9. Hyperparameter Optimization

The surrogate models were trained using 188 unique combinations of binary and unary operators, as outlined in subsection 4.3.4. Given the extensive number of combinations, performing hyperparameter optimization could help identify trends in the preferred training settings. The data was first sorted by MSE and the five cases with the lowest values were selected. If the difference in MSE between the lowest and the fifth-lowest case was less than 5%, additional cases were included to ensure a minimum 5% increase in MSE. This was the case for the cyclopentane SR1W model, were the six lowest MSE cases are considered. The results of the hyperparameter optimization for the SR1E and SR1W models, trained on the complete dataset with a maximum expression complexity of 60, are summarized in Table E.5 and E.6.

Table E.5: Hyperparameter optimization for SR1E models

Working fluid	hat	sin	cos	tan	exp	log
Ethanol	1	4	1	3	3	3
R134a	3	4	3	2	2	2
Butane	2	5	2	-	2	4
Cyclopentane	1	1	3	3	1	4
Toluene	3	4	4	4	2	3
MM	4	4	4	3	-	3

Table E.6: Hyperparameter optimization for SR1W models

Working fluid	hat	sin	cos	tan	exp	log	abs
Ethanol	1	2	2	4	1	2	5
R134a	2	3	4	4	3	3	5
Butane	4	2	1	2	3	4	5
Cyclopentane*	1	4	4	3	1	2	6
Toluene	2	1	4	5	4	1	5
MM	4	2	3	4	1	2	5