

Intelligent Control of Refrigeration Systems: Adaptive Fuzzy Strategies for Superheat Regulation

Aakash Raj Panneerselvam

Offshore Energy Systems
OES4,
Spring Semester 2025

Master Thesis





AALBORG UNIVERSITY

STUDENT REPORT

Title:

Intelligent Control of Refrigeration Systems: Adaptive Fuzzy Strategies for Superheat Regulation

Theme:

Master Thesis

Project Period:

Spring Semester 2025

Project Group:

OES4

Participants:

Aakash Raj Panneerselvam

Supervisors:

Zhenyu Yang

yang@energy.aau.dk

Roozbeh Izadi-Zamanabadi

riz@es.aau.dk

Page Numbers: 61

Date of Completion:

1st September 2025

Abstract:

The evaporator system is very important for refrigeration and air conditioning systems because it makes sure that heat is exchanged efficiently. But its performance is affected by nonlinear behavior, especially when the system gain is very large in some operating situations. One of the biggest problems in controlling the evaporator system is keeping the superheat levels just right. If the superheat is too low, liquid refrigerant could go into the compressor, which could break the system. Too much superheat, on the other hand, makes cooling less effective, which wastes energy. So, it's very important to be able to control superheat levels exactly. To solve this problem, two advanced control methods are suggested: Adaptive Fuzzy-Based Sliding Mode Control This method uses a rule-based system to dynamically change control settings to keep stability and performance, even when there are nonlinearities. Adaptive Fuzzy PID Control with Adaptive Gains , this is a hybrid method that uses fuzzy logic to constantly adjust the proportional and integral gains, making sure the system works best in all situations. The goal is to create an adaptive control technique that makes the superheat at the desired temperature while successfully handling changes in gain and delays in delivery. The suggested control system must strike a compromise between efficiency and system protection by keeping superheat to a minimum and stopping liquid from getting into the compressor. Choosing the right control gains, fixing transport delay problems, and changing control rules on the fly are all important design choices that will help the evaporator system run smoothly and efficiently.

Contents

List of Figures	v
List of Tables	vii
Preface	1
1 Introduction	2
1.1 Refrigeration Systems	2
2 Problem Analysis	5
2.1 Motivation	5
2.1.1 Practical Limitation	6
2.1.2 Optimization Problems Infeasibility	6
2.1.3 Negative Bias in Temperature Regulation	6
2.2 Need for Improved Weight Relationships	7
2.2.1 Potential for Shorter Prediction Horizons	7
2.3 Model and Control Limitation	7
3 Modeling	8
3.1 Pressure Enthalpy Diagram	8
3.2 Transfer function Model	9
3.3 Transfer function of the evaporator	9
3.4 Transfer function of the expansion valve	10
3.5 Superheat Temperature Transfer Function	11
3.6 Normalization of Transfer Functions	11
3.7 Actual System Modeling Description	12
3.7.1 Temperature Superheat Reference	15
4 Control Strategies	17
4.1 Traditional PID controller	18
4.2 Fuzzy PID Control	19
4.2.1 Fuzzy PID Architecture	19

4.2.2	Knowledge Base	23
4.2.3	Defuzzification	24
4.2.4	Implementation of Fuzzy PID	27
4.2.5	Fuzzy PID Block Structure	27
4.2.6	Normalization of Controller Inputs	28
4.2.7	Scaling Factors in Fuzzy PID Design	28
4.3	Adaptive Fuzzy PID	32
4.3.1	Implementation of Sliding Mode Controller	42
4.3.2	Adaptive Fuzzy Sliding Model Control	44
5	Simulation and Results	50
5.1	Results Using Transfer Function	50
5.2	Reference Tracking Performance	50
5.2.1	Adaptive <i>SMC</i> control	50
5.2.2	Adaptive Fuzzy <i>PID</i>	52
5.3	Adaptive Fuzzy <i>SMC</i>	52
5.4	Total Simulation	53
5.5	Adaptive Fuzzy <i>PID</i> with different rules	53
6	Conclusion	58
6.1	Future Work	58
	Bibliography	59

List of Figures

3.1	Pressure Enthalpy Graph	9
3.2	Two-Phase Zone	12
3.3	Two-Phase Zone With Parameters	13
3.4	Cascade Control for T_{SH} and T_{air}	16
4.1	Conventional PID Controller	18
4.2	Architecture of Fuzzy	20
4.3	Features of Membership Function	21
4.4	Max Membership Method	25
4.5	Center of Gravity Method	26
4.6	Fuzzy PID Block Structure	28
4.7	Level Membership Function	30
4.8	Output membership functions for valve control actions.	30
4.9	Architecture Of Adaptive Fuzzy <i>PID</i>	32
4.10	Temperature Superheat Error	34
4.11	Rate of Change of Error	35
4.12	Universe of Discourse for Gain K'_p	39
4.13	Universe Of Disclosure for Gain K'_d	40
4.14	Universe Of Disclosure for Alpha	41
4.15	Sliding Mode Control	45
4.16	Block Diagram of Adaptive Fuzzy <i>SMC</i>	46
5.1	Adaptive <i>SMC</i> control of Temperature Superheat	51
5.2	Adaptive <i>SMC</i> control Valve Opening Degree	51
5.3	Adaptive Fuzzy <i>PID</i> of Temperature Superheat	52
5.4	Adaptive Fuzzy <i>PID</i> of the Valve OD	52
5.5	Adaptive Fuzzy <i>SMC</i> of <i>TSH</i> and the Valve OD	53
5.6	All Controllers System Reference	53
5.7	All Controllers System Response	54
5.8	All Controllers Valve Actuation	54
5.9	Valve Opening Degree	55

5.10 Superheat Temperature 56
5.11 Universe of discourse for rate of change of superheat error 56
5.12 Universe of discourse for Valve 57

List of Tables

4.1	Map of Fuzzy Rules	30
4.2	Fuzzy Tuning Rules for K'_p of Valve Opening Degree	36
4.3	Fuzzy Tuning Rules for K'_d of Valve Opening Degree	36
4.4	Fuzzy Tuning Rules for α of Valve Opening Degree	37
4.5	Rule Table	40
4.6	Triangular Membership Function Parameters	47
4.7	Triangular Membership Function Parameters	47
4.8	Fuzzy Rules for Switching Gain η	48

Preface

The idea for this project was initiated by Danfoss and later developed in collaboration with the company. I would not have been able to carry out this work without the invaluable support of my supervisors, Zhenyu Yang (Aalborg University) and Roozbeh Izadi-Zamanabadi (Danfoss). I am sincerely grateful to Zhenyu for his continuous guidance, for taking extra time to help me move forward whenever I was stuck, and for connecting me with Roozbeh. I would also like to thank Roozbeh for his generous support, for sharing his expertise in refrigeration systems, for providing a simulation model, and most importantly, for giving me the opportunity to work on this project. Finally, I wish to thank my friends and family for their constant encouragement and support throughout this journey.

Aalborg University, 1st September 2025

Aakash Raj Panneerselvam
<apanne23@student.aau.dk>

Chapter 1

Introduction

1.1 Refrigeration Systems

Commercial and Industrial refrigeration systems are among the most significant energy-consuming industries. They account for 15% of global electric energy consumption and are expected to grow over the next few decades, significantly influencing CO₂ emissions. [1]

Cooling systems are essential to various industries such as air conditioning, food storage, and pharmaceuticals. Most of these systems function through a vapor compression cycle, which includes four primary elements: the compressor, condenser, expansion valve, and evaporator. The proper operation of these elements is crucial for enhancing energy efficiency and reducing environmental impact. In this context, the evaporator system is vital as it absorbs heat from the surrounding environment, thus providing the necessary cooling effect [2].

Heat absorption by the evaporator is one of the most crucial steps in a vapor compression refrigeration cycle. The evaporator is the component of the refrigeration system through which the refrigerant fluid absorbs heat from the surrounding medium to create the desired cooling, since the refrigerant fluid exists as a vapor after leaving the expansion device. The refrigerant's thermodynamic condition inside the evaporator determines how well this heat transfer mechanism works. The heat transfer coefficient between the evaporator surface and refrigerant is significantly higher in the two-phase (liquid vapor) state than at a state where the refrigerant has completely vaporized. Therefore, to achieve the desired cooling capacity at the evaporator, it is better for the evaporator to consist of some mixture of liquid and vapor refrigerant instead of completely vaporized refrigerant fluid [3].

There can, however, be an important operational constraint in this situation. Although sending a two-phase refrigerant (a mixture of liquid and vapor refrigerant) will enhance

the heat transfer in the evaporator, a refrigerant that is entering the compressor must be vaporized. Allowing liquid refrigerant to enter the compressor can create conditions for severe mechanical failure due to liquid slugging, when incompressible liquid droplets hit the moving parts of the compressor. Liquid slugging will reduce the service life of a compressor and could even cause immediate failure. Hence, the skill of the system operation is to balance the operation to obtain maximum cooling capacity in the evaporator with providing vaporized refrigerant to the compressor [4].

To bring balance, engineers apply a particular parameter called superheat, which indirectly measures the liquid fraction of the fluid in the evaporator. Superheat is the temperature difference between the refrigerant fluid at the outlet of the evaporator relative to the saturation temperature of the refrigerant fluid at that pressure [2]. In practice, this is done by measuring the inlet and outlet temperatures of the refrigerant. If the refrigerant at the outlet is warmer than its saturation temperature, it means the refrigerant is fully vaporized and has been heated slightly above its boiling point, ensuring the compressor is within its safe operating limits [4].

Superheat of just about 5K is a common compromise used in practice for refrigeration. Studies throughout the years have indicated this value is a reasonable compromise between maximizing system operation and minimizing the likelihood of damaging the compressor. However, the actual minimum temperature could vary greatly across system design, operation, and the evaporators themselves. Every evaporator has an intrinsic feature known as a minimum stable super-heating level. This is where the control system continuously overshoots and undershoots, ultimately reducing the efficiency of operation and responding with an unstable behavior. This is another objective to not allow the setpoint for super-heating to be set too low [4].

The expansion valve is the part of the cycle that connects the high-pressure side and the low-pressure side. It controls the flow of refrigerant into the evaporator. The typical expansion valves used in real time applications are capillary tubes and thermostatic expansion valves (*TXVs*). Both devices limit the flow in terms of refrigerant metering control; however, they have limitations. Capillary tubes are easy to make and cheap but have no flexibility, and *TXVs* now have more sophistication than capillary tubes but they still have a lag in their response time and cannot operate effectively over a wide range of operating conditions [4].

In refrigeration systems with highly variable operating conditions, expansion devices are usually calibrated for worst-case conditions. This is done for safety, but at the expense of efficiency. When the expansion valve is set to maintain too high a degree of superheat for worst-case conditions, the system may run with significantly elevated superheat levels for extended periods during normal operating conditions. This is a significant amount

of time when the refrigerant being delivered is not optimal and therefore is impacting system efficiency in a sizable way. Tassou and Al-Nizari (1993) illustrated this efficiency loss using empirical studies. They showed that from 8K super-heating to 14K resulted in a difference of approximately 9% in the coefficient of performance (COP) of a commercial refrigeration system. The results emphasized the importance of accurate superheat control. Even insignificant increases can have a measurable effect on the performance [4].

The maintenance of an optimal superheat temperature is crucial for enhancing the operational efficiency of refrigeration systems and averting potential compressor damage. The control of superheat can be achieved by adjusting the degree of valve opening [2]. Proper superheat control prevents liquid refrigerant from entering the compressor, which could otherwise cause mechanical failure. In addition, it improves the energy efficiency of the system and stabilizes its thermal performance under varying operating conditions. Controlling the superheat temperature in refrigeration systems is a complex task, primarily due to the nonlinear behavior inherent in evaporator dynamics and the sensitivity of the system to varying operational conditions. The relationship between the valve opening degree and superheat is highly nonlinear, which makes it difficult to achieve accurate and stable control using conventional methods. Moreover, refrigeration systems often operate under fluctuating thermal loads, varying ambient temperatures, and inconsistent refrigerant flow rates, all of which introduce disturbances that significantly affect system behavior. These changing conditions make it difficult to maintain optimal superheat levels and can lead to performance degradation or even system instability if not properly managed. While traditional *PI* and *PID* controllers are widely used, their performance heavily depends on accurate tuning, which is complicated by the diversity of system designs and operating environments. As such, there is a growing need for adaptive and intelligent control methods that can respond dynamically to system changes without relying on detailed mathematical models. Addressing these challenges is essential for improving the efficiency, reliability, and safety of modern refrigeration systems [5].

Chapter 2

Problem Analysis

2.1 Motivation

Traditional control techniques, although commonly implemented, frequently struggle to tackle the intricacies and variations present in these operations. Traditional controllers like Proportional-Integral-Derivative (*PID*) controllers are often the preferred method of control for their general simplicity, ease of implementation, and longevity. But these controllers often make significant errors in the presence of nonlinear dynamics, time-varying processes, and disturbances. The failure of (*PID*) controllers to respond to these nonlinearities tends to lead to less efficient, and suboptimal operation [2]. Moreover, this limitation contributes to higher energy consumption, which poses a significant challenge in modern refrigeration applications where sustainability and energy efficiency are critical priorities [6].

To tackle these challenges, a joint research project has been started by Aalborg University and Danfoss to introduce complex adaptive control techniques into refrigeration systems. The aim of this project is to improve the temperature control of the evaporator subsystem which is a crucial subsystem managing the cooling process. As part of this work, an adaptive control strategies for an existing model of the evaporator has been formulated with the aim of simulating and mimicking its operating behavior to an extent of accuracy. The model has not only been created for simulation but also as a predictor to support the development and validation of control strategies.

Fundamentally, this model of evaporator is a structure that converts the physics and dynamics of the system into mathematical expressions. Predictions are performed on system parameters, focusing on the cooling air temperature T_{air} , since it is a critical quantity to predict which plays a key role in system efficiency and user comfort. To accomplish this, the model combines historical system data with input control signals, thus making it possible for it to predict future states of the system with some level of

reliability [7].

The approach used in building this model is that of Gaussian Process Modeling *GPM*. *GPM* is a capable statistical method that enables one to develop a probabilistic model from a relatively short amount of observed data points. As an alternative to deterministic models, Gaussian Processes offer a flexible framework for understanding complex system dynamics and quantify prediction uncertainty. This feature makes *GPM* especially suitable for systems like refrigeration, with very dynamic operating conditions, for which deterministic models only might not be enough [7].

The existing system demonstrated expected performance under simulated conditions:

- Mean Squared Error (MSE) as low as 0.003°C
- Tracking accuracy with maximum deviation of $\pm 0.27^{\circ}\text{C}$
- Ability to maintain stable air temperatures in test scenarios

The MPC-based strategy has a number of practical limitations despite these results, especially when taking into account real-time deployment and long-term system flexibility [7].

2.1.1 Practical Limitation

Using a Gaussian Process model to implement MPC presented significant difficulties that compromise its applicability for plug-and-play industrial systems [7].

2.1.2 Optimization Problems Infeasibility

In approximately 10% of the scenarios, the Model Predictive Control (*MPC*) formulation yielded infeasible solutions [7]. This problem could be caused by the model's linearization and the use of a lengthy prediction horizon, which could result in circumstances in which the model is unable to identify a workable solution for the control inputs.

2.1.3 Negative Bias in Temperature Regulation

The cooling air's temperature could be efficiently controlled by the *MPC*, although its predictions showed a negative bias. This bias is probably caused by a large input-change weight (R) relative to the error-weight, which can influence how aggressive the control technique is [7].

2.2 Need for Improved Weight Relationships

It might be possible to eliminate the negative bias by implementing a more aggressive control method by modifying the weight relationship in the cost function. This indicates that the current weight settings may not be optimal for achieving the desired control performance.

2.2.1 Potential for Shorter Prediction Horizons

One idea to tackle the challenges of optimization problems is to work with a shorter prediction horizon. This tweak might help the MPC stay feasible and boost its responsiveness to shifts in the system dynamics.

2.3 Model and Control Limitation

The GP-based *MPC* in the current model is significantly dependent on linearization and adequately representative training datasets. Although, the evaporator system is non-linear, varies over time, and is influenced by various disturbances including changes in refrigerant properties, flow variations, and fluctuating loads. Multiple operational factors restrict the effectiveness of Model Predictive Control *MPC* in regulating the superheat temperature in an evaporator, even though it excels in systems with clearly defined dynamics and constraints [7]. Under field condition it is common for missing of critical sensor data. For example, pressure sensors and inlet water temperature sensors are prone to calibration drift, electrical noise, mechanical failure, or communication loss. These measurements are crucial for the computation of thermodynamic properties such as saturation temperature, enthalpy variation, or superheat margin. The absence of these measurements may result in the model generating erroneous predictions and control interventions. In these circumstances, conventional controllers demonstrate an inability to adapt to novel operating conditions, as they do not possess a mechanism for re-tuning the control gains. This deficiency culminates in an incapacity to uphold optimal control, thereby resulting in suboptimal system performance characterized by increased overshoot, prolonged settling times, and, in certain instances, unstable system dynamics [7]. The failure to sustain effective regulation of superheat under such limitations presents dangers not only to system efficiency but also to the safety of the hardware specifically the compressor, which may incur damage due to the ingress of liquid refrigerant during instances of evaporator flooding [7]. To address these challenges, the implementation of adaptive fuzzy logic control strategies can enhance the system's ability to maintain optimal superheat levels despite the inherent uncertainties and nonlinearities present in evaporator dynamics.

Chapter 3

Modeling

3.1 Pressure Enthalpy Diagram

The pressure-enthalpy diagram, shows the refrigerant starting its journey through the vapor compression refrigeration (*VCR*) cycle at point 1, which is when the refrigerant enters the evaporator. During the transition between point 1 and 2, the refrigerant absorbs heat from the surrounding environment, which causes it to change from a two-phase mixture at the inlet of the evaporator to a superheated vapor at the outlet point [8].

From point 2 to 3 the refrigerant is in a superheated state as it is compressed, causing a little increase in pressure. By the time the refrigerant reaches a state at point 3, it is now a high-pressure, high-temperature vapor that has entered the condenser. In the condenser the refrigerant then rejects heat to the outside environment. As a result, after the refrigerant leaves the condenser, which is represented by point 4, it is at a lower temperature, typically in a saturated liquid or sub-cooled liquid state [8].

First, refrigerant passing between point 4 and point 1 goes through the expansion valve. In this unit operation, the refrigerant experiences a sudden loss of pressure. After the expansion process, the refrigerant then enters the evaporator as a two-phase liquid and vapor mixture at point 1 meaning that the cycle is complete [8].

By incorporating electronic expansion valves (*EEVs*) and variable-speed compressors, modern *VCR* systems can provide significantly more flexibility by implementing advanced control strategies to promote system efficiency and performance. In these applications, the control inputs are related to how far as the *EEV* is opened, which controls the refrigerant mass flow rate at the evaporator inlet (\dot{m}_{in}), by controlling the compressor speed during compressor operation, which controls the mass flow rate at the outlet (\dot{m}_{out}), and also, through the speed of the evaporator fan, which controls the airflow rate over the evaporator [8].

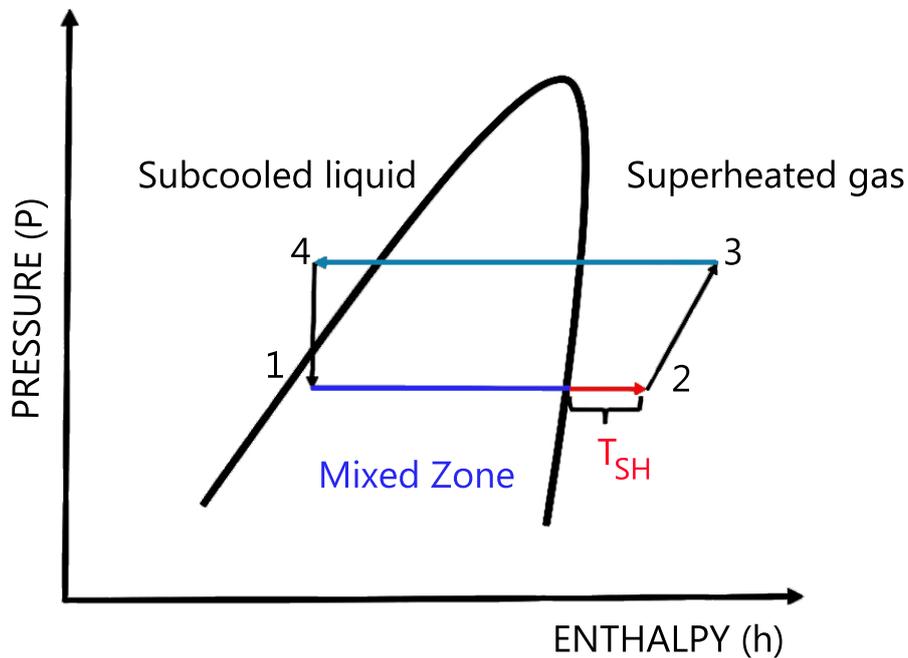


Figure 3.1: Pressure Enthalpy Graph

3.2 Transfer function Model

Before the actual process of implementing control mechanism within the existing system model, there is a need for a comprehensive analysis that involves not only the modeling of the valve and evaporator but also the temperature superheat transfer function, all of which is done to ensure that the entire system operates properly.

3.3 Transfer function of the evaporator

The dynamic behavior of the evaporator was obtained experimentally by applying step excitation of the refrigerant mass flow rate at its inlet and recording the corresponding variations in the superheat at the outlet. The resultant superheat response was fitted to the (FOTS) First Order Time Delay model. The ratio between the inlet and outlet of the

evaporators are put together as an open loop transfer function, which can be expressed as a first order time delay.[9]

$$G_E(p) = \frac{\Delta T_s(p)}{\dot{m}_r(p)} = \frac{-K_E}{1 + \theta p} \exp\{-\tau p\} \quad (3.1)$$

Here:

$\Delta T_s(p)$ - Temperature Superheat

$\dot{m}_r(p)$ - mass flow rate

The above parameters vary with respect to the evaporation temperature and compressor speed. Empirically, over various operating conditions the values for K_E , h and q were expressed using regression analysis as a function of operating conditions. For control implementation, the exponential delay term was approximated with a first order term. [9]

Now, the Transfer function of the evaporator is expressed as:

$$G_E(p) \approx -\frac{K_E}{(1 + hp)(1 + qp)} \quad (3.2)$$

Here:

K_E Evaporator gain

h Time constant

q Transport delay

[9]

It is modeled as a second-order system characterized by a negative gain, indicating that an increase in mass flow initially results in a decrease of superheat owing to transient flooding conditions.

3.4 Transfer function of the expansion valve

The main function of the expansion valve is to regulate the mass flow rate entering the evaporator to ensure smooth operation without hunting. The valve exhibits consistent mass flow characteristics as a function of the stepper motor's position, which operates the expansion valve. Through experimentation, it was determined that the correlation between the mass flow rate and the stepper motor position is linear, allowing for the proportional gain K_v to be derived from the slope of the flowposition characteristic.

The expansion valve transforms the valve opening control signal $u(t)$ into the refrigerant mass flow $\dot{m}_r(p)$ and is conventionally depicted as a first-order system or, as an integrator.

The corresponding transfer function of the expansion valve is expressed as:

$$G_V(p) = \frac{\dot{m}_r(p)}{U(p)} \quad (3.3)$$

Here:

$U(p)$ - Control Signal

$\dot{m}_r(p)$ - mass flow rate

3.5 Superheat Temperature Transfer Function

The dynamic behavior of superheat within the evaporator can be articulated through the subsequent transfer function:

$$G_S(p) = \frac{1.5e^{-10s}}{50s + 1} \quad (3.4)$$

The Superheat Transfer Function was presented by "Professor Roozbeh". The superheat sensor is characterized as a first-order low-pass filter with a minor time delay and negative polarity, thereby providing the measured superheat $SH_{measured(t)}$.

3.6 Normalization of Transfer Functions

The system is scaled by normalization form with the listed transfer functions of expansion valve, evaporator and superheat. Once the normalization is done the transfer function looks like

$$G_{\text{plant}}(s) = \frac{0.194925}{2500s^4 + 800s^3 + 65s^2 + s} \quad (3.5)$$

The above transfer function can also be written as

$$G_{\text{plant}}(s) = \frac{0.915}{s(50s^2 + 15s + 1)(50s + 1)} \quad (3.6)$$

3.7 Actual System Modeling Description

Evaporators within refrigeration systems employ a vapor-compression cycle mechanism that facilitates the transfer of heat. The fundamental principle involves enabling the refrigerant to flow through the evaporators, wherein the heat exchange occurs between the fluid circulating within the evaporator and the ambient air temperature. This process is critical as it allows the refrigerant to absorb heat, transitioning from a liquid to a vapor state, thus providing the necessary cooling effect in the system. To attain the necessary superheat temperature, the superheat must exceed the temperature of the evaporator. This ensures that the refrigerant is fully vaporized before it enters the compressor, preventing potential damage and enhancing cooling efficiency [7].

The existing model is comprised of two section:

- Mixture Zone
- Superheat Zone

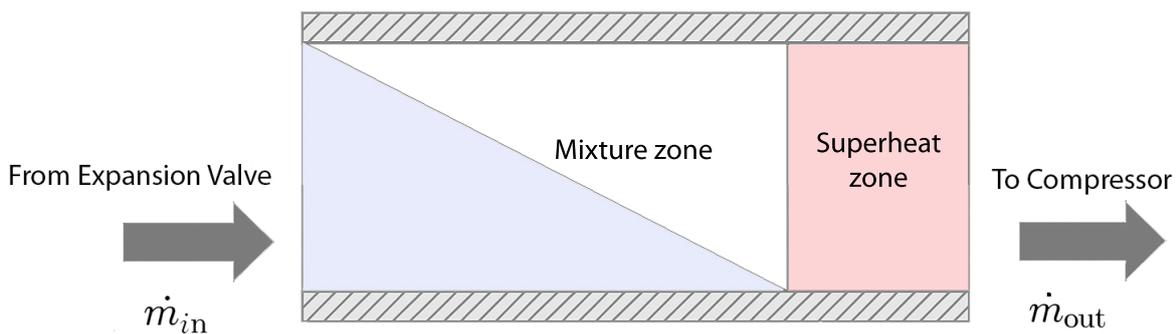


Figure 3.2: Two-Phase Zone

In the mixture zone, the liquid refrigerant is transitioning from liquid to vapor and then enters the Superheat Zone. The process of evaporation takes place as the refrigerant in the zone is in the transition state that is being vaporized from the liquid state to the gas state [7].

A more detailed diagrammatic representation of the Evaporator system is depicted, the diagram is taken for more detailed thermodynamical equations.

The dynamic temperature changes of the walls of the mixture zone describe the temperature change of the mixture zone.

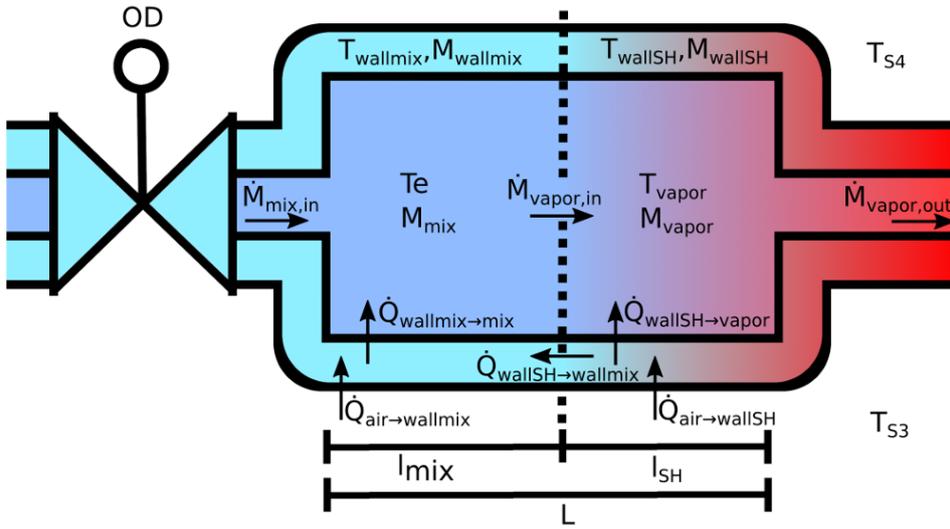


Figure 3.3: Two-Phase Zone With Parameters [7]

$$\dot{T}_{\text{wallmix}} = \frac{\dot{Q}_{\text{air-wallmix}} + \dot{Q}_{\text{wallSH-wallmix}} - \dot{Q}_{\text{wallmix-mix}}}{M_{\text{wallmix}} c_{p,\text{wall}}} \quad (3.7)$$

Taking into account the dynamics of the valve, the mathematical representation of the mass flow of the refrigerant flowing through the valve is expressed as:

$$\dot{M}_{\text{mix,in}} = OD \cdot K_v \sqrt{(\rho_{\text{liq}}(1 - \gamma) + \rho_{\text{gas}}\gamma)(P_{\text{rec}} - P_e)} \left[\frac{\text{kg}}{\text{s}} \right] \quad (3.8)$$

where:

- OD is the valve opening degree in percent [.]
- K_v is the valve flow constant $\left[\frac{\text{kg}}{\text{s}\sqrt{\text{Bar}} \text{ kg m}^{-3}} \right]$
- ρ is the density of the medium $\left[\frac{\text{kg}}{\text{m}^3} \right]$
- γ is the void mean fraction [.]
- P is the pressure [Bar]
- Δh_{lg} is the specific latent heat $\left[\frac{\text{kJ}}{\text{kg}} \right]$

The governing mass flow refrigerant equations represent the pressure difference, which is determined by the upstream pressure (P_{rec}) and downstream pressure (P_e) is the driving force for the refrigerant transport into the evaporator. A pressure difference flow of high-pressure sub-cooled liquid refrigerant, which swiftly expands through the valve opening. Consequently, a low-pressure, two-phase refrigerant mixture flows into the evaporator, initiating the heat absorption process essential for phase change [7].

The valve Opening and closing should be governed by valve transfer function equation:

$$OD = \frac{OD_{CTRL}}{\tau \cdot s + 1} [7] \quad (3.9)$$

where:

- OD_{CTRL} Control Signal
- τ Valve Time Constant

The variation of wall temperature within the superheat zone is governed by dynamic heat transfer equations that model energy exchange between the wall and the refrigerant vapor. In contrast to the mixture zone, the wall experiences heat loss or gain due to conduction and mass flow where the superheat zone exhibits a similar mechanism, along with an inverted sign for the conduction terms. In effect, heat that is absorbed in one region may correspond to a reduction in the other, ensuring conservation of energy across the system [7].

The below equations characterize the rate of temperature change at the wall surface, which is influenced by the heat energy transferred from the vapor during evaporation[7].

$$\begin{aligned} \dot{T}_{\text{wallSH}} &= \frac{\dot{Q}_{\text{air-wallSH}} - \dot{Q}_{\text{wallSH-wallmix}} - \dot{Q}_{\text{wallSH-vapor}}}{M_{\text{wallSH}} C_{p\text{wall}}} \left[\frac{K}{s} \right] \\ \dot{T}_{\text{vapor}} &= \frac{\dot{Q}_{\text{wallSH-vapor}} - \dot{Q}_{\text{vapor-vapor,in}}}{M_{\text{vapor}} C_{p\text{vapor}}} \left[\frac{K}{s} \right] \end{aligned} \quad (3.10)$$

In order to calculate the mass of the refrigerant vapor in the superheat zone by referring to the look-up table. Based on the look-up table, the given parameters such as evaporator pressure P_e and the vapor temperature, the corresponding vapor density can be retrieved. Once the density is identified, the mass of vapor within the superheated region is calculated by multiplying this value by the spatial volume occupied by the vapor [7].

The volume of the evaporator given by [7]:

$$V = \frac{M_{\text{mix, max}}}{\rho_{\text{liq}}(P, T)(1 - \gamma) + \rho_{\text{gas}}(P, T)\gamma} \equiv \text{constant} \quad (3.11)$$

The mass of vapor is given by [7]:

$$M_{\text{vapor}} = (V - V_{\text{mix}}(P, T)) \rho_{\text{gas}}(P, T) \quad (3.12)$$

where:

V_{min} volume occupied by the mixture in the evaporator.

When determining the evaporator's volume distribution, a parameter involving the maximum mass mixture which represents a fully flooded state. This parameter is used to estimate the total volume of the evaporator. By subtracting the volume occupied by the two-phase mixture from this total, the remaining volume available for superheated vapor is obtained [7].

$$V_{\text{mix}}(P, T) = \frac{M_{\text{mix}}}{\rho_{\text{liq}}(P, T)(1 - \gamma) + \rho_{\text{gas}}(P, T)\gamma} \quad [\text{m}^3] \quad (3.13)$$

Once the volume of superheated vapor within the evaporator has been determined, the corresponding length of the superheat zone can be calculated:

$$l_{SH} = \frac{V - V_{\text{mix}}(P, T)}{V} L \quad [\text{m}] \quad (3.14)$$

[7]

3.7.1 Temperature Superheat Reference

According to the Gaussian Process Model, evaporator's performance and compressor's safety depend on maintaining a suitable T_{air} for cooling while guaranteeing safer superheat levels T_{SH} respectively. The primary goal of the evaporator in this Gaussian process model is to cool the air to a specific temperature, T_{air} , and to maintain an evaporator superheat of 8K, which is 8K above the evaporator temperature.

Cascade Control

The dual objective supports the cascade control methodology. The outer loop calculates the difference between the the desired air temperature $T_{\text{air,ref}}$ and the measured air temperature T_{air} . The error is fed to the *PI* controller providing the reference for superheat $T_{SH,ref}$. A target for the inner loop that makes the actual superheat T_{SH} follows these

reference values is $T_{SH,ref}$. Using a PI controller, this inner loop calculates the necessary valve opening degree based on the error terms.

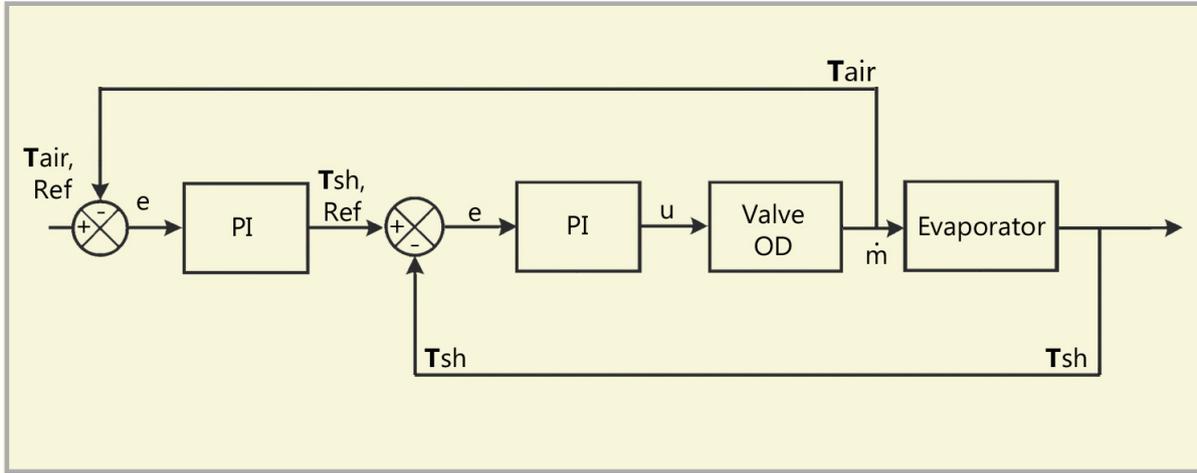


Figure 3.4: Cascade Control for T_{SH} and T_{air}

The Valve *OD* is used as a shared actuator between T_{air} and T_{SH} . Within the cascade control framework, the T_{air} is controlled indirectly by manipulating the superheat T_{SH} . When $T_{air} > T_{air,ref}$, the outer-loop controller decreases $T_{SH,ref}$; in contrast, when $T_{air} < T_{air,ref}$, the outer-loop controller increases $T_{SH,ref}$. After receiving this reference, the inner-loop controller constantly compares T_{SH} to $T_{SH,ref}$ and adjusts the expansion valve opening degree *OD* in an effort to reduce errors. As $T_{SH,ref}$ decreases, more refrigerant flows, increasing the mixture zone and improving latent heat transfer, which lowers T_{air} . In contrast, raising $T_{SH,ref}$ causes the heat transfer area to reduce, limit refrigerant flow, and raise T_{air} . The outer loop controls thermal performance through this hierarchical structure, while the inner loop maintains refrigerant safety and dynamic tracking. The single actuation of the Valve will result in the influence of both T_{air} and T_{SH} . The defrost function eliminates ice buildup by injecting additional heat into the controller.

To ensure a safe operation, boundary layers are implemented which is the saturation bounds where T_{air} maintained within 5°C and T_{SH} 8K above the evaporator temperature

Chapter 4

Control Strategies

The provided GP Model uses a Linear Model Predictive Control (*LMPC*) framework to develop control strategies for regulating the air temperature T_{air} effectively. The primary objective of this control is to regulate the Air Temperature T_{air} by adjusting the opening degree of the valve OD [7].

The dynamic behavior of the system is represented using Gaussian Process Model that allows for the predictive mechanism. The GP model's predictive capabilities, is designed to estimate how variations in the valve position will affect air temperature over an extended period of time. GP model establishes a correlation between the measured air temperature and the previous valve setting, thereby providing significant understanding of the system's dynamics. The implementation of Gaussian Process Modeling enables a more accurate prediction of the air temperature in refrigeration systems. While GP model offers insights about the behavior of the system, its computing complexity and intrinsic non-linearity make it less appropriate for real-time control implementation. To mitigate this limitation, the GP Model is linearized at a specific operating condition which will reflect the normal behavior of the system [7].

Prior to implementing Adaptive Controllers in GP Model, it is essential to develop, implement and validate Non-Adaptive Controllers using the transfer function mentioned in the modelling section. This provides the foundation for comprehending the dynamic properties of the system and their response to different inputs. This allows the control design to be analyzed more easily. With this foundational structure in place, the transition to Adaptive Controllers can be carried out with certainty to ensure improved performance and stability in regulating the superheat temperature. The adaptive controllers might leverage real-time data to adjust control parameters dynamically, enhancing the system's responsiveness to varying operational superheat conditions, for which implementation of control strategies needs to be done.

4.1 Traditional PID controller

In Industrial applications such as HVAC, *PID* controllers remains the choice of preference die to their simple design, easy implementation and robust operations. Although *PID* have been working around for decades, their popularity has continued to grow significantly because of the advancement in tuning techniques that make it easier to adapt the controller to different process requirements and improve overall system performance. One such tuning technique is the Ziegler-Nichols method [10]. With the Ziegler-Nichols technique, the Proportional-Integral-Derivative (*PID*) controller was tuned using the derived transfer functions which is mentioned in section 3.6.

This process makes it simpler to tweak the *PID* parameters to improve the system's stability and responsiveness. Furthermore, tuning the controller significantly improves system operations by reducing overshoot and optimizing settling times, ensuring proper regulation of the superheat temperature.

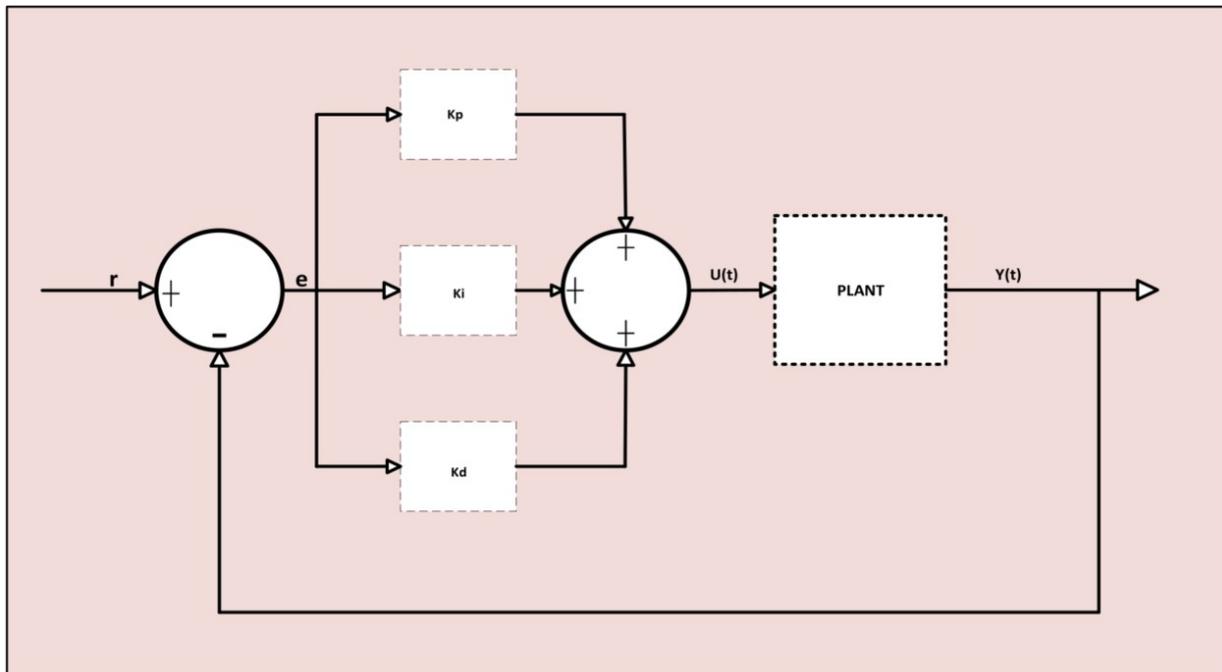


Figure 4.1: Conventional PID Controller

The gain values obtained are

- $K_p = 0.01$
- $K_i = 0.1166e-7$

- $K_d = 0.001$

which were determined to achieve a desirable balance between responsiveness and stability in controlling the superheat temperature.

4.2 Fuzzy PID Control

Most of the work was carried out using MATLAB Toolbox named Fuzzy Logic Designer, which allows the development of fuzzy inference systems to model and control the superheat temperature. The Fuzzy Logic Designer enables the creation of fuzzy rules and membership functions, facilitating enhanced control over superheat temperature dynamics.

4.2.1 Fuzzy PID Architecture

The structure of the fuzzy control system comprises four principal components:

- the Fuzzification
- The knowledge base,
- Inference mechanism
- the Defuzzification

This design allows for effective handling of the nonlinear dynamics inherent in superheat control [11].

The configuration of a fuzzy controller requires tuning of four parameters:

- **Determination of the Sampling Time:** The sampling interval of the controller is selected with respect to the plant model.
- **Normalization of the gain parameters:** The normalization of the gain parameters involves scaling the input variables to ensure that the fuzzy controller operates across different operating conditions.
- **Formulation of rules:** The formulation of rules involves the relationship between the inputs and the outputs of the fuzzy inference system.
- **Specification of Membership Functions:** For the non linear relationship between the the inputs and the outputs, appropriate membership functions shapes are defined to capture the system's behavior

[2]

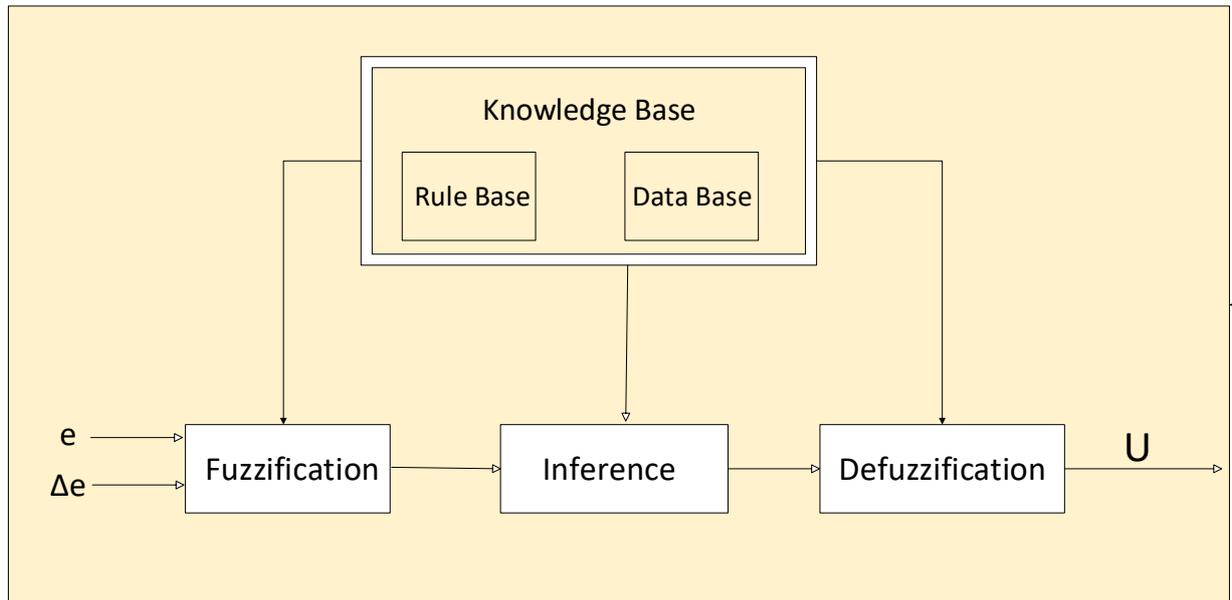


Figure 4.2: Architecture of Fuzzy

The diagram presented above illustrates the implementation of the fuzzy logic controller.

Fuzzification Process

The first process is the fuzzification of input variables. These inputs are termed as crisp values. The process of fuzzification is to convert the crisp numerical inputs into fuzzy values in fuzzy sets, allowing the controller to interpret the input. Since digital systems typically process information in binary form, they are not inherited to interpret vague or uncertain data directly. To address this, crisp inputs must be translated into linguistic variables such as "low," "moderate," and "high," or Small, Zero and Big which can be evaluated within the framework of fuzzy logic [12].

Depending on the system, the linguistic variables may differ. Assigning membership values to the crisp inputs considered to be a step in the fuzzification process. This conversion method involves designating the appropriate membership functions to correlate the crisp inputs with fuzzy sets, which enables the controller to process information in a way that resembles human reasoning. [12].

In fuzzy logic notation, this relationship is typically expressed using the Greek letter μ [12].

There are many techniques for constructing the membership functions. The characteristics of these functions influence which rules are activated, the strength of the activation, and the smoothness of the controller's output response.

There are 3 key features of membership function:

- Core
- Boundary
- Support

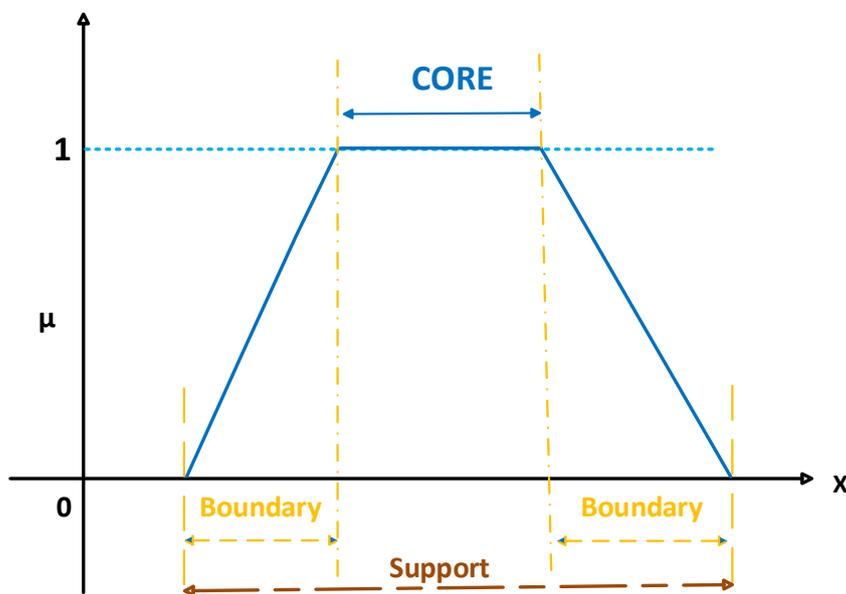


Figure 4.3: Features of Membership Function

Generally, The membership assumes values ranges defined by the user with respect to the system.

Consider a Fuzzy Set A which can be represented by pair

$$A = \{x \mid \mu(x) \mid x \in X\} \quad A = \{(x, \mu(x)) \mid x \in X\}$$

Here,

- **A** - Fuzzy Set
- **x** - Value
- μ - Membership Function
- ϵ Universe of Disclosure

Core

This lets the controller make smart choices based on linguistic variables. This ensures confident decision by defining values that are full membership in the set A [12].

$$A = \{x \mid \mu(x) = 1\}$$

Support

The support of a fuzzy set A comprises of every element in the universe of discourse that has a degree of membership greater than zero. If an element has any degree of relation with the set, even it is small, it is associated with the support [12].

$$A = \{x \mid \mu_A(x) > 0\}$$

Boundary

For fuzzy set A, the range of all elements in the set is between 0 and 1 in the region of universe x, but the membership function is incomplete [12].

$$A = \{x \mid 0 < \mu_A(x) < 1\}$$

Along with the 3 key features of the membership function there are other features such as cross-over points, normality, classification of fuzzy sets which put together will shape the controllers response, robustness to noise and stability.

It is evident that selection and design of appropriate membership function is important as they determine which inputs are construed in accordance with fuzzy principles. In practice, membership functions can take upon various shapes such as triangular, trapezoidal, Gaussian, or bell-shaped with each shape holding Its own advantages depending on the nature of the system and its performance requirement [12].

A triangular membership function is defined by straight line segments with three points: starting at the minimum of zero, reaching a maximum of one, and settling again at zero. With three parameters, it is easy to frame and tune the rules, making it a straightforward and preferred choice for a membership function.

A trapezoidal function takes the shape of a trapezoid, defined by four points, which allows for flexibility in modeling. Behavior at the edges are considered crucial. The waveform in the function takes a linear ramp and a flat top. When mapping the membership functions even when the inputs exceed the modelled ranges the rule is preserved and not deviated improving the robustness. The choice of membership functions directly influences the controller's performance, as trapezoidal functions are often preferred for their flexibility and ease of implementation in fuzzy systems.

Systems which require dynamic variation in a shorter time span, it is preferred to use a triangular or trapezoidal waveform.

In cases where good accuracy, smooth peaks, and easier tuning are required, the Gaussian or sigmoid waveform is preferred. However, it has a disadvantage: long-tailed structures may activate distant rules slightly [13]. With respect to varying operating conditions the selection of appropriate membership functions, the appropriate selection of membership functions is essential for optimal control performance, as they have a direct influence on the system's responsiveness and managing nonlinear dynamics [14]

4.2.2 Knowledge Base

The Fuzzy Inference system is known to be the core component in the Fuzzy Logic System. The knowledge base includes data bases with membership functions and fuzzy rules that are essential for guiding the fuzzy inference system in processing inputs to achieve the desired control outcomes. Decision making mechanism is an important part in the Inference system, the rules are formulated as antecedent (IF) and the consequent (THEN) parts [15]. These rules are generally described in a manner similar to how humans make decisions in the presence of uncertain information. These rules represents expert knowledge or intuitive reasoning regarding the relationship between inputs and the desired outputs [2].

There are 2 most commonly used Inference Method:

Mamdani Inference Method:

Mamdani systems are defined that the rules are formulated in a specific way. The Mamdani inference method employs fuzzy logic principle in both the antecedent (IF) and consequent (THEN) parts of the fuzzy rules. The underlying principle involves representing both the inputs and outputs where the representation of the output remains in fuzzy set unless defuzzification process is applied. The Mamdani inference method's linguistic transparency facilitates better understanding and interpretation of fuzzy rules, making it a popular choice in various applications of fuzzy logic control. Its rules are similar to the way of human-reasoning delivering intuitiveness that enhances the interpretability of the control system, making it easier to adjust and tune the rules as needed.

In practice, Mamdani works well with non-linear systems, however dealing with large number of rules and detailed membership functions requires intensive computation [13].

TakagiSugeno (TS) Inference Method:

Unlike the Mamadni method, the T-S Inference generates crisp output by using the linear combination as a mathematical function of input variables. The Takagi-Sugeno method's avoids generating fuzzy set as an output rather respond numerically thus reducing the complexity of the computation involved [13]. The mathematical approach of this Inference system favors with data-driven approach, thus enabling integration with neural network learning and optimization based parameter tuning. Together, these characteristics improve the system's ability to learn and adapt [13].

Despite of this benefits, the use of mathematical approach would lead to challenges to derive the input-output fuzzy relationship. T-S Method would have trouble working with Non-Linear systems because the model assumes the the relationship between inputs and outputs is linear. Therefore, careful consideration must be taken into account while selecting the inference method for non-linear systems.

In theory, the membership functions of a fuzzy sets can assume wide variety of forms, but considering in real-time applications, when computational efficiency is crucial, a uniform membership function shape is preferred. There are bell shaped, triangular, trapezoidal and exponential shapes are used whereas trapezoidal and triangular membership functions are commonly favored due to their due to their simple parametric representation, minimal computational overhead, and ease of implementation [15]. Here, the two inputs for the system are transformed into Linguistic Variables.

4.2.3 Defuzzification

The last step in the Fuzzy Logic Control is the is the defuzzification process, which converts the fuzzy output from the inference engine into a actionable crisp value that will be used as output control signal from the controller. Defuzzification can also be referred to as the "rounding off" method [12]. There are various techniques for defuzzification such as

- (1) Max-membership principle,
- (2) Centroid method,
- (3) Weighted average method,

- (4) Meanmax membership,
- (5) Center of sums,
- (6) Centre of largest area,
- (7) First of maxima or last of maxima

[12]

Each has its own distinct characteristics, which work well according to the application of the system and the choice of inference method used.

Max-membership principle

This method is also termed a height method. The Max-Membership function selects the output value(s) as which the function achieves its maximum [12].

It can be expressed mathematically by,

$$\mu_A(y^*) > \mu_A(y)$$

Here, y is the defuzzified output.

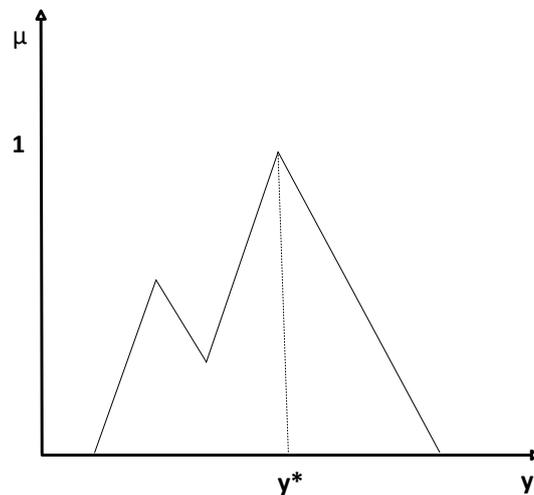


Figure 4.4: Max Membership Method

In the Inference mechanism the rule firing stage involves computing the firing strength by combining the the membership values $\mu_A(y)$ associated with each input variable using logical operator. Furthermore, the firing strength determines the degree of activation for each fuzzy rule and impacts the formulation of the output fuzzy set during the aggregation process. At the point where the aggregated fuzzy output set exhibits the highest degree of membership, the defuzzification out is determined. This method is preferred where quick approximation is sufficient [16]

Center of Gravity (COG) Method

After the fuzzy inference, the result of the aggregated output membership function, that represents the combination of all activated rules as the output. This approach identifies the center point of the area under the output membership function, Mathematically, it involves computing the weighted average of all possible output values, weighted by their corresponding membership degrees. Mathematically, this involves calculating the weighted average of all possible output values, where the weights correspond to their respective membership degrees [16]

$$y^* = \frac{\int \mu_{\bar{A}}(y) \cdot y \, dy}{\int \mu_{\bar{A}}(y) \, dy}$$

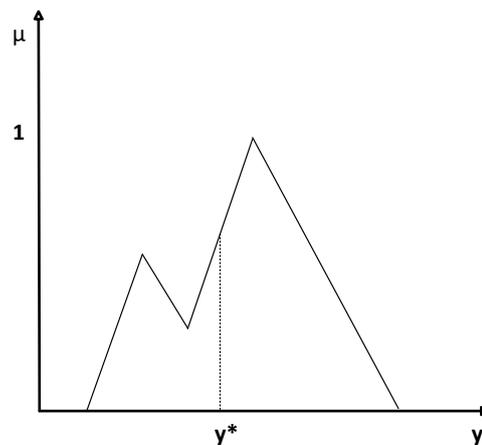


Figure 4.5: Center of Gravity Method

The COG method is the mostly preferred method for defuzzification. This preference for the Center of Gravity (COG) method is based on its effectiveness in producing smooth and accurate output values, making it particularly suitable for applications involving fuzzy logic control [17]

Weighted average method

The weighted average method is applicable only when the output membership functions are symmetrical, as it cannot effectively handle asymmetrical distributions. In this approach, each output membership function is assigned a weight based on its maximum membership value, and the final crisp output is calculated as the weighted average of these values [12]

Mathematically it can be represented as

$$y^* = \frac{\sum \mu_{\tilde{A}}(\bar{y}_i) \cdot \bar{y}_i}{\sum \mu_{\tilde{A}}(\bar{y}_i)}$$

4.2.4 Implementation of Fuzzy PID

As mentioned in section 3.2 the implementation goal is to achieve good reference tracking performance for the transfer function model. This is done to test the performance of non-adaptive controllers with constant gains. The equation 3.6 represents the 4th order transfer function that includes the behavior of the evaporator, superheat and the valve dynamics. From the transfer function, the denominator polynomial represents a factor s that indicates the presence of integrator in the plant, Nonetheless, it suggests that any controller containing integral action must be carefully tuned to avoid destabilizing the system.

4.2.5 Fuzzy PID Block Structure

The Fuzzy *PID* can be designed or a black box structure which is readily available can be used, rather than the conventional *PID* form. Here, the a black box structure is utilized for control implementation. The structure of a fuzzy *PID* was deigned in a parallel configuration manner. The strength of fuzzy *PID* controller's architecture uses the positive aspects of proportional-integral *PI* and proportional-derivative *PD* control action under the fuzzy framework, while overcoming the complexities involved in designing and tuning a three-dimensional usually fuzzy rule required in traditional fuzzy *PID* designs [18]. In accordance with each branch, the *PD* branch controls the transient response characteristics, especially the rise time and overshoot reduction, while the *PI* branch rectifies the error in order to attain zero steady-state deviation from the setpoint [18].

Figure 4.6: Fuzzy PID Block Structure

4.2.6 Normalization of Controller Inputs

According to the conventional design, there are two inputs, and both branches share the same inputs. The inputs to the fuzzy inference system must be normalized to fit within the fuzzy domain, typically defined over the interval $[-1, 1]$. Two primary input signals are used:

- **Normalized Error** $E(k)$

$$E(k) = C_e \cdot e(k) \quad (4.1)$$

- **Normalized Rate of Change of Error** $\dot{Y}(k)$

$$\dot{Y}(k) = -C_d \cdot \frac{dy(k)}{dk} \quad (4.2)$$

Here,

- $e(k)$: the tracking error between the reference $r(k)$ and actual output $y(k)$.
- C_e and C_d : normalization gains selected to scale the error and rate of change of error into the fuzzy domain.

The derivative term is based on the change in system output rather than the change in error. This design prevents *derivative kick*, a phenomenon where sudden changes in the reference signal cause large spikes in the derivative term.

Now, the change in error can also be denoted in terms of output difference:

$$\dot{Y}(k) = -(y(k) - y(k - 1)) \quad (4.3)$$

Thus, the controller only responds to changes in the plant output, not to sudden deviations in the setpoint.

4.2.7 Scaling Factors in Fuzzy PID Design

In order to ensure that the fuzzy PID controller exhibits dynamic behavior similar to that of a conventional PID controller, it is necessary to scale the constants C_e , C_d , C_0 , and C_1 . The gain values K_p , K_i , and K_d of the conventional PID are obtained through the Ziegler–Nichols tuning method, which facilitates the adjustment of these constants to optimize

the fuzzy PID controller's performance [19].

Here,

- C_e is set to 0.1 to ensure that the error input falls within the fuzzy logic domain.
- C_d is linked to the derivative path and ensures consistency between fuzzy and classical derivative action:

$$C_d = \frac{2K_i C_e}{K_p - \sqrt{K_p^2 - 4K_i K_d}} \quad (4.4)$$

[19]

Based on these scaling factors, the values of C_0 and C_1 are derived as:

$$C_0 = C_e K_i, \quad C_1 = C_d K_d \quad (4.5)$$

[19]

Fuzzy Inference System Design

The fuzzy inference system (FIS) employs a Mamdani inference method that includes Gaussian and triangular membership functions to capture nonlinearities in the control behavior.

Input Membership Functions

The inputs are described using linguistically defined fuzzy sets:

$$\text{Level } (e) : \{\text{Low, Ok, High}\}, \text{Rate } (\dot{Y}) : \{\text{Negative, None, Positive}\}$$

Output Membership Functions

The output of the FIS corresponds to the control actions required to regulate the valve. The output variable **valve** is defined by five fuzzy sets:

- Close Fast
- Close Slow
- No Change
- Open Slow
- Open Fast

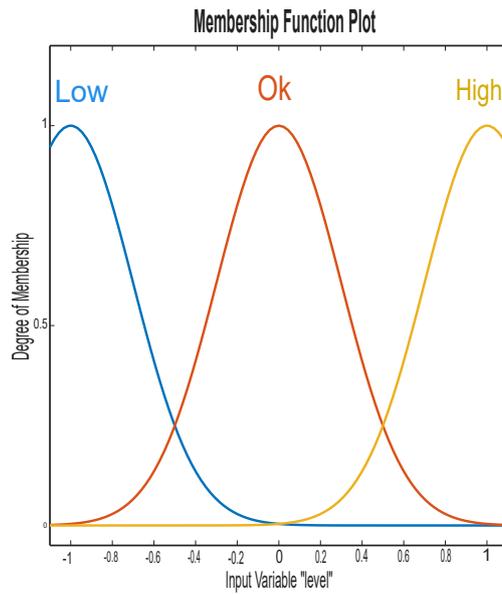


Figure 4.7: Level Membership Function

Figure 4.8: Output membership functions for valve control actions.

Rule Base

The formulation of rules was based on expert knowledge and research conducted on evaporator systems [17]. By evaluating the inputs (*level* and *rate*) alongside the response of the control action, the valve's behavior was analyzed to assist in the formulation of the rules.

A lookup table of fuzzy rules was developed as shown in Table 4.1.

Table 4.1: Map of Fuzzy Rules

Error (Level)	Rate Negative	Rate Zero	Rate Positive
Low	Open_Fast	Open_Fast	Open_Fast
Okay	Open_Slow	No_Change	Close_Slow
High	Close_Fast	Close_Fast	Close_Fast

The fuzzy rule base is also expressed explicitly as:

- Rule 1: If **level** is *okay*, then **valve** is *no_change*.
- Rule 2: If **level** is *low*, then **valve** is *open_fast*.

- Rule 3: If **level** is *high*, then **valve** is *close_fast*.
- Rule 4: If **level** is *okay* and **rate** is *positive*, then **valve** is *close_slow*.
- Rule 5: If **level** is *okay* and **rate** is *negative*, then **valve** is *open_slow*.

The result of rule evaluation is an aggregated fuzzy output that describes the control response under current operating conditions [17].

Defuzzification

In the defuzzification process, the Center of Gravity (COG) method is applied due to its accuracy and smoothness [17]. The fuzzy outputs are combined and defuzzified into a single normalized control effort $u_f(t)$.

The fuzzy controller output is expressed as:

$$u(t) = u_{PI}(t) + u_{PD}(t) \quad (4.6)$$

$$u(t) = C_0 \int u_f(t) dt + C_1 u_f(t) \quad (4.7)$$

where:

- C_0 : the output gain associated with the PI branch
- C_1 : the gain associated with the PD branch

[19]

Gain Scheduling Behavior

Classical PID controllers operate with fixed gains, making them less responsive in systems with varying dynamics [4]. In contrast, the fuzzy logic controller adapts its control action based on real-time input conditions. This adaptive adjustment is referred to as **gain scheduling**, where the controller modifies its effective gains dynamically according to the error and rate of change of error [4].

4.3 Adaptive Fuzzy PID

It is a known fact that the *PID* controllers are commonly used industrial control because of its simplicity, intuitive design, and efficiency across many operating circumstances. Nevertheless, conventional *PID* controllers often struggle with nonlinearities, delays and time-varying dynamics, leading to less than optimal performance in complex systems such as refrigeration [20]. Introducing Adaptivity approach into the conventional *PID* method can significantly improve the performance by managing the challenges of non linear dynamics and varying operational conditions, since *PID* controllers uses constant gains and manually tuned values [20]. The adaptability of the varying parameters is executed using an adjustment mechanism, enabling real-time modifications to the control parameters and improving the ability of the system to respond to changing superheat condition.

The expression for the *PID* control can be written as

$$u(k) = K_p e(k) + K_i T_s \sum_{i=1}^n e(i) + \frac{K_d}{T_s} \Delta e(k) \quad (4.8)$$

T_s is the sampling time frequency.

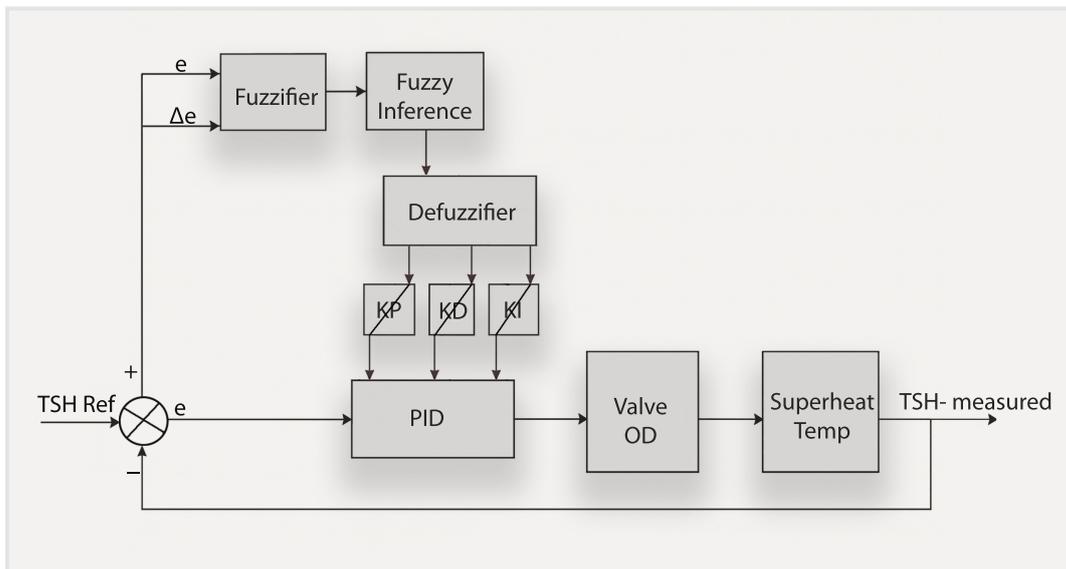


Figure 4.9: Architecture Of Adaptive Fuzzy *PID*

Fuzzification

The fuzzification process converts temperature errors and changes in error values into fuzzy sets. This allows the controller to make well-informed choices based on linguistic variables and membership functions [21]. Here, The fuzzy controller modifies the conventional *PID* by applying fuzzy reasoning to control the *PID* parameters. In accordance with the design specifications, the fuzzy controller applied for *PID* parameter tuning is formulated with two input variables and three corresponding outputs. In this adaptive fuzzy *PID* framework, the two primary inputs are the error in the superheat temperature, denoted as " e ", and the rate of change of error, denoted as " Δe " [22].

$$e(k) = T_{SH\text{REFERENCE}} - T_{SH\text{MEASURED}} \quad (4.9)$$

$$\Delta e(k) = e(k) - e(k - 1) \quad (4.10)$$

In this case, seven membership functions were employed to facilitate the conversion between crisp values into fuzzy values. This number is widely recognized for systems exhibiting significant nonlinearity. Declaring membership function less than seven comes with its own compromise of the responsiveness, whereas incorporating with more than seven membership can lead to higher computational complexity and affect real-time implementation [17]. The fuzzy inputs are characterized by assigning membership functions with linguistic terms such as negative large (*NL*), negative medium (*NM*), negative small (*NS*), zero (*ZE*), positive small (*PS*), positive medium (*PM*) and positive large (*PL*) [23]. The domain of disclosure for superheat error and rate of change of superheat error is set within a range of [-1 1].

The controller gains are normalized using the following linear transformation for convenience :

$$K'_p = \frac{K_p - K_{p\text{min}}}{K_{p\text{max}} - K_{p\text{min}}} \quad (4.11)$$

$$K'_d = \frac{K_d - K_{d\text{min}}}{K_{d\text{max}} - K_{d\text{min}}} \quad (4.12)$$

In this frame work, the adjustment of *PID* parameters relies on the inputs of Temperature Superheat error $e(k)$ and rate of change of error Δe . To simplify the tuning process, the integral time constant is linked proportionally to the derivative time constant [23].

$$T_i = \alpha * T_d \quad (4.13)$$

Here, α is termed as the scaling factor. The dimensionless gain ratio, α , indicates how strong the integral action is in comparison to the derivative action [23]. Now, the integral gain can be reformulated as

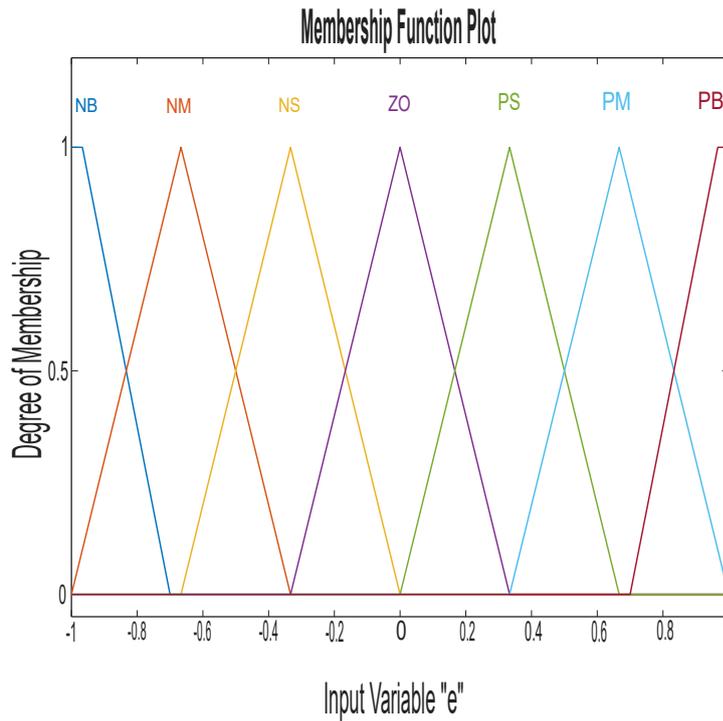


Figure 4.10: Temperature Superheat Error

$$K_i = \frac{K_p}{\alpha T_d} = \frac{K_p^2}{\alpha K_d} \quad (4.14)$$

Since in the time constant $T_d = \frac{K_d}{K_p}$

Fuzzy Rules

The primary objective is to maintain the superheat temperature close to the setpoint while minimizing overshoot, oscillations, and control effort. Following the approach of Zhao et al. The fuzzy rules were designed based on the step response experimentation [23]. Fuzzy rules were established for different operational phases of the system by analyzing the dynamic properties of superheat temperature in regard to changes in valve position. A significant effort from the controller is necessary during the initial transient phase to achieve a rapid rise time and quickly, guide the system towards the superheat reference. Thus, gain value K_p should be large enough to achieve fast reaction whereas, gain value of K_d should be small in order to avoid suppressing the initial action [23].

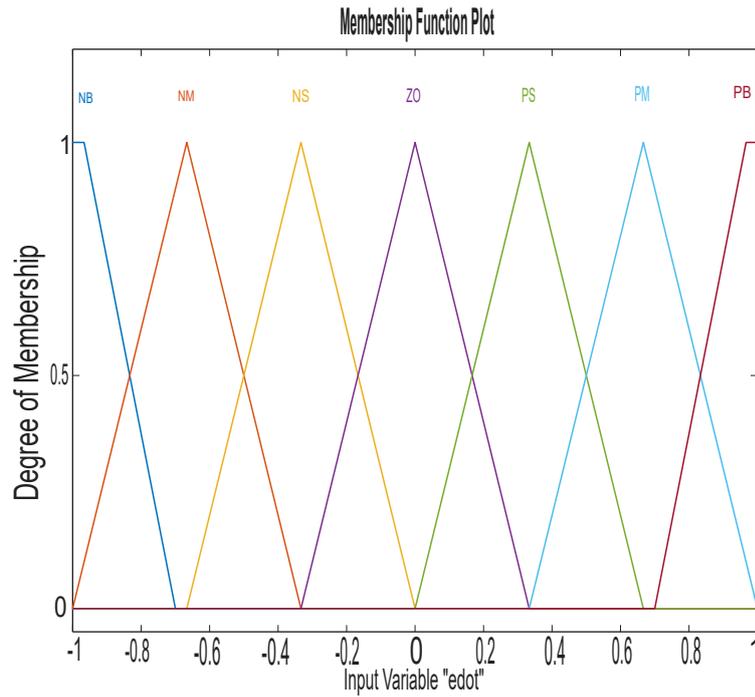


Figure 4.11: Rate of Change of Error

This behavior is formulated as fuzzy rules

$$\text{If } e(k) \text{ is PB and } \Delta e(k) \text{ is ZO, then } K_p \text{ is Big, } K_d \text{ is Small, } \alpha = 2 \quad (4.15)$$

As mentioned earlier above, the relative strength is the ratio between integral and derivative time constant which will compute the K_i gain values. Here based on the constant values of α aggressiveness of the control is decided.

- When α value ranges between 1 to 2 the integral action is stronger and promotes faster steady state convergence [23].
- When α value ranges 3 and more than 3 the integral action is weaker and reduces overshooting [23].

Now, any further increases should be stopped by the control during the overshoot period. In this region, the controller should use a smaller K_p , a larger K_d , and a weaker integral action. This scenario occurs when the superheat approaches steady state [23].

This behavior is formulated as fuzzy rules

$$\text{If } e(k) \text{ is ZO and } \Delta e(k) \text{ is NB, then } K_p \text{ is Small, } K_d \text{ is Big, } \alpha = 5 \quad (4.16)$$

These rules, when combined, form a set of 49 fuzzy rules with seven linguistic levels for input errors and the rate at which they change. These rules are represented in a table

$e(k)$	$\Delta e(k)$						
	NB	NM	NS	ZO	PS	PM	PB
NB	B	B	B	B	B	B	B
NM	S	B	B	B	B	B	B
NS	S	S	B	B	B	B	S
ZO	S	S	S	B	S	S	S
PS	S	S	B	B	B	B	S
PM	S	B	B	B	B	B	S
PB	B	B	B	B	B	B	B

Table 4.2: Fuzzy Tuning Rules for K_p' of Valve Opening Degree [23]

$e(k)$	$\Delta e(k)$						
	NB	NM	NS	ZO	PS	PM	PB
NB	S	S	S	S	S	S	S
NM	B	B	S	S	S	B	B
NS	B	B	B	B	B	B	B
ZO	B	B	B	B	B	B	B
PS	B	B	B	B	B	B	B
PM	B	B	S	S	S	B	B
PB	S	S	S	S	S	S	S

Table 4.3: Fuzzy Tuning Rules for K_d' of Valve Opening Degree [23]

Defuzzification

The defuzzification process converts fuzzy values from the inference system into crisp values because the controller cannot process fuzzy values. The fuzzy inference system calculates the truth value μ of each rule, also known as the degree of membership function, by multiplying the membership values from the inputs.

$$\mu_i = \mu_{A_i}(e(k)) \cdot \mu_{B_i}(\Delta e(k)) \quad (4.17)$$

Here,

- μ_{A_i} membership function values for error

$e(k)$	$\Delta e(k)$						
	NB	NM	NS	ZO	PS	PM	PB
NB	2	2	2	2	2	2	2
NM	3	3	2	2	2	3	3
NS	4	3	3	2	3	3	4
ZO	5	4	3	3	3	4	5
PS	4	3	3	2	3	3	4
PM	3	3	2	2	2	3	3
PB	2	2	2	2	2	2	2

Table 4.4: Fuzzy Tuning Rules for α of Valve Opening Degree [23]

- μ_{B_i} membership function values for rate of change of error

[23]

The output membership function consists of two curves which are represented by "Big" B or "Small" S . While operating at a particular point when the need for the valve opening degree to be large the gain is on the B curve and vice versa when the gains are need to be small.

The output membership functions are defined using two linguistic terms: "Big" B and "Small" S , each corresponding to a specific fuzzy set represented by a distinct and independent curve. During operation, when the control system requires a bigger valve opening, the associated gain corresponds with the "Big" membership function. In contrast, when a lesser control effort is sufficient, the gain aligns with the "Small" membership function [23].

All values in the fuzzy inference system are scaled between 0 and 1, as it works with normalized inputs.

$$\sum_{i=1}^n \mu_i = 1 \quad (4.18)$$

The normalized gains K'_p and K'_d can be calculated through defuzzification using the center of gravity approach, based on the values of μ_i [23].

$$K'_p = \sum_{i=1}^m \mu_i K'_{p,i} \quad (4.19)$$

$$K'_d = \sum_{i=1}^m \mu_i K'_{d,i} \quad (4.20)$$

$$\alpha = \sum_{i=1}^m \mu_i \alpha_i \quad (4.21)$$

The values of K'_p and K'_d that correspond to the grade of the membership function μ_i for the i^{th} rule are denoted by $K'_{p,i}$ and $K'_{d,i}$, respectively [23]. Upon defuzzification, the normalized values are interpolated within the specific ranges to compute the actual values of K_p , K_i and K_d . From the equations 4.15 and 4.14 the gains are computed using following formulas:

$$K_p = (K_{p,\max} - K_{p,\min}) \cdot K'_p + K_{p,\min} \quad (4.22)$$

$$K_d = (K_{d,\max} - K_{d,\min}) \cdot K'_d + K_{d,\min} \quad (4.23)$$

$$K_i = \frac{K_p^2}{\alpha \cdot K_d} \quad (4.24)$$

Here, the fuzzy logic controller's normalized outputs are denoted by K'_p and K'_d . The minimum and maximum acceptable values for K_p are indicated by $K_{p,\min}$ and $K_{p,\max}$, while those for K_d are represented as $K_{d,\min}$ and $K_{d,\max}$.

The permissible lower and upper bounds for *PID* controller gains can be found by observing the system's behavior. To determine these limits, a commonly used technique called the Ziegler-Nichols (Z-N) tuning method was used [23]. Two significant values are determined using this procedure K_u and T_u these values are obtained from the existing Gaussian Process Model.

These are the suggested ranges for the gains:

$$K_{p,\min} = 0.32K_u, \quad K_{p,\max} = 0.60K_u \quad (4.25)$$

$$K_{d,\min} = 0.08K_u T_u, \quad K_{d,\max} = 0.15K_u T_u \quad (4.26)$$

Here, T_u - Time of oscillation which was obtained through trial and error method.

Numerous simulations and research were carried out to establish the basis for the formulation of the given bounded values [23].

Although there are several comparable techniques to construct the Adaptive Fuzzy *PID*, these are one method for fuzzy gain scheduling. The implementation of the fuzzy rules is the only modification. The rule clearly makes direct use of raw profits gains.

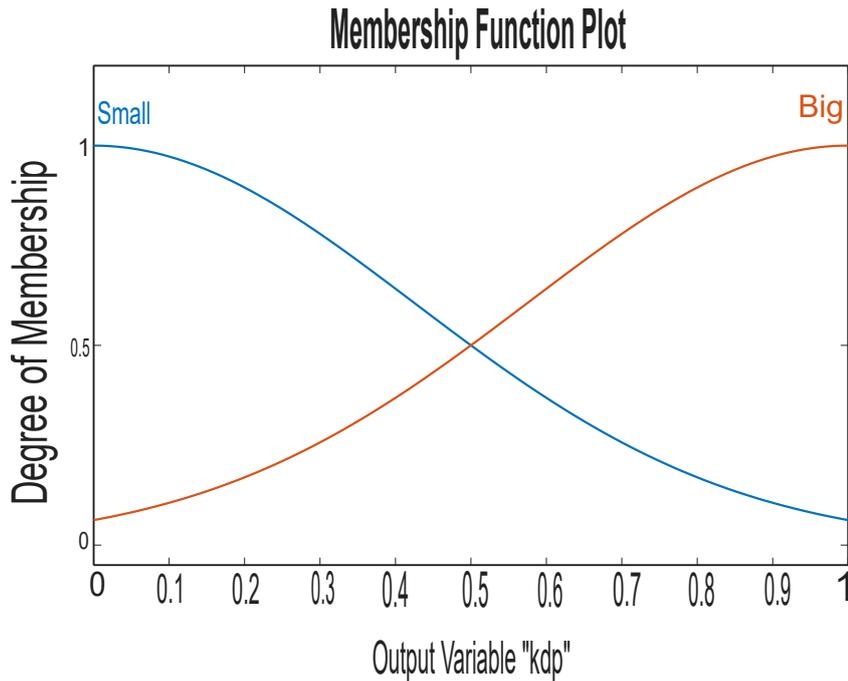


Figure 4.12: Universe of Discourse for Gain K'_p

In this case, the rules are formulated as

If $e(k)$ is A and $\Delta e(k)$ is B , then U is C

The well-defined "IF-Then" statements that link the valve response to the system fuzzy conditions comprise the rule basis. Each rule correlates the fuzzy input conditions with corresponding control actions, which are defined by the knowledge base, that is essential for the fuzzy controller.

These relationships are expressed through a set of linguistic rules, for example :

- If the superheat error is negative large (NL) and the rate of change of superheat error is Negative Large (NL), this indicates that the superheat temperature is very low and dropping drastically, which would lead to flood-back. Now, the controller must respond immediately and the valve should be closed [2].
- If the Superheat error is Negative Medium (NM) and the the rate of change of superheat error is Negative Large (NL), this indicates a slower drop in the temperature of the superheat which is still a issue. Now the controller must respond immediately and the valve should be closed [2].

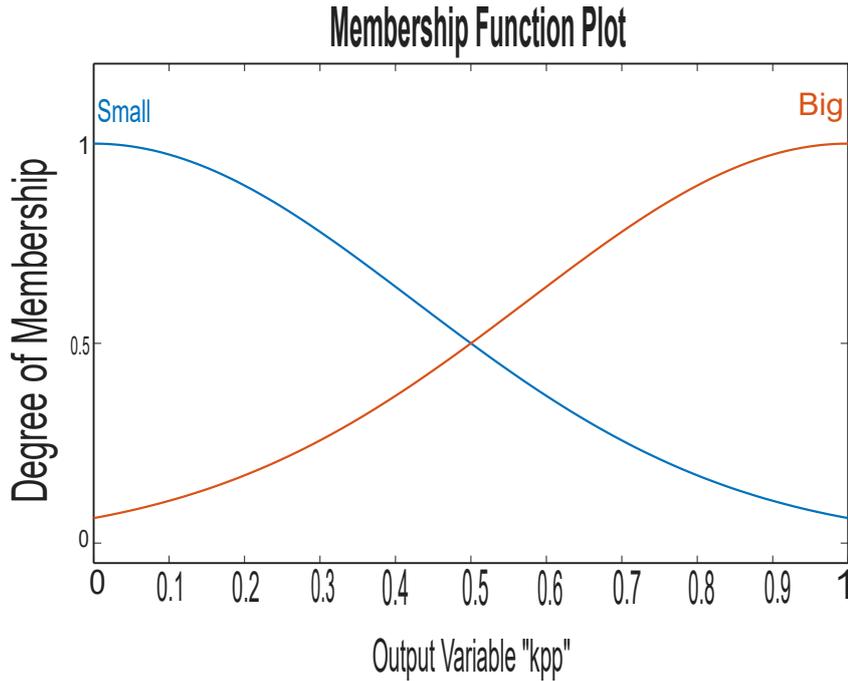


Figure 4.13: Universe Of Disclosure for Gain K'_d

e	Δe (change in error)						
	NL	NM	NS	Z	PS	PM	PL
NL	NL	NL	NL	NM	NS	Z	PS
NM	NL	NL	NM	NS	Z	PS	PM
NS	NL	NM	NS	Z	PS	PM	PL
Z	NM	NS	Z	PS	PM	PL	PL
PS	NS	Z	PS	PM	PL	PL	PL
PM	Z	PS	PM	PL	PL	PL	PL
PL	PS	PM	PL	PL	PL	PL	PL

Table 4.5: Rule Table

- If the superheat error is Zero (ZE) and the rate of change of superheat error is Zero (ZE), then system is stable, and no action is required on the valve [2].

Similarly all 49 rules are framed based on this domain knowledge of the temperature of the superheat. The fuzzy rules considers the reasoning into control logic which acts as a catalyst to frame the rules respective to Fuzzy PID . Unlike traditional PID controllers, which depend on fixed parameters, fuzzy inference uses linguistic mappings to dynamically understand the system state and modify the control output [2].

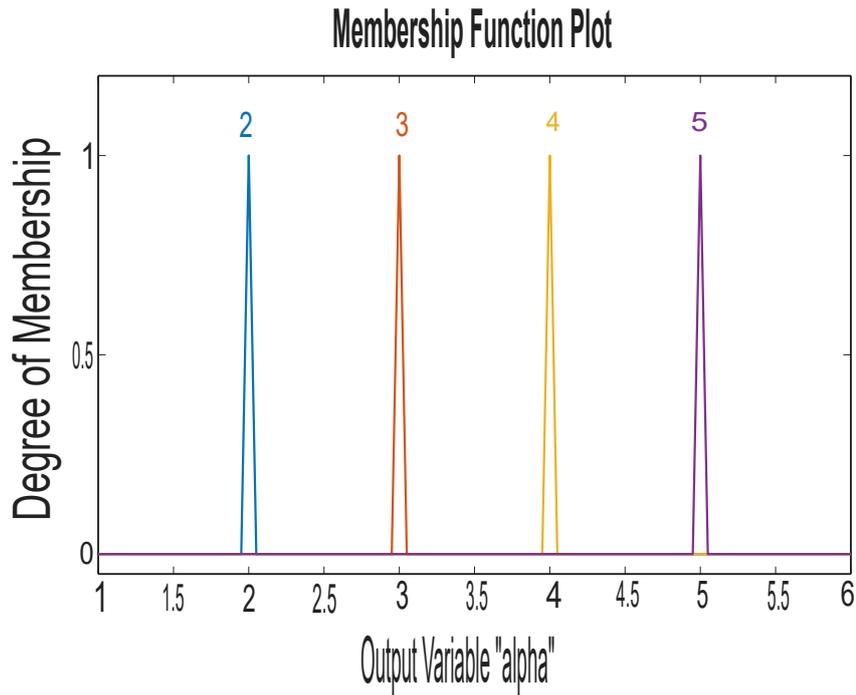


Figure 4.14: Universe Of Disclosure for Alpha

The controller operates by continuously monitoring the data from the Model and calculates for error (e) the rate of change of error Δe . A fuzzy inference system processes these inputs to generate an output that provides the corrective action, in this case the change in the valve's opening percentage [2]. Here, the current valve opening dynamically scales the fuzzy controller's output based on the following relation:

$$u'(t) = \Delta u'(t) \cdot u(t) + u(t) \quad (4.27)$$

Here,

- $u(t)$: current percentage opening of the electronic expansion valve (EEV).
- $\Delta u'(t)$: output from the fuzzy inference system.
- $u'(t)$: updated valve opening.

When the valve is fully open, which indicates a high evaporator load, the formulation allows the controller to adjust. The fuzzy-PI control in this case allows for larger corrective actions. In contrast, the fuzzy-PI control permits smaller adjustments when the valve is almost closed, indicating a low evaporator load. The control output can be scaled according to the current valve position, which allows the system to react to various load scenarios [2].

4.3.1 Implementation of Sliding Mode Controller

Sliding Mode Control (*SMC*) is considered a robust control method that can effectively handle disturbances and uncertainties [24]. A variable structure control technique is implemented for the given plant model. The design of the *SMC* controller occurs in two steps. The first step involves defining a sliding surface that represents the desired system behavior, ensuring stability under varying conditions. Subsequently, the control law is designed to drive the system state towards this surface, compensating for uncertainties and disturbances.

In (*SMC*), it is essential to derive a state–space representation of the model [24]. However, in Sections 3.6 and 3.7, the model is represented as a fourth–order transfer function.

State–Space Representation

The state–space representation is expressed as:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (4.28)$$

$$y(t) = Cx(t) + Du(t) \quad (4.29)$$

with the state vectors defined as:

$$x(t) = [y(t) \quad \dot{y}(t) \quad \ddot{y}(t) \quad y^{(3)}(t)]^T, \quad x_d(t) = [y_d(t) \quad \dot{y}_d(t) \quad \ddot{y}_d(t) \quad y_d^{(3)}(t)]^T$$

The controllable canonical form is applied to obtain the matrices:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -0.32 & -0.026 & -0.0004 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix},$$

$$C = [0.00036596 \quad 0 \quad 0 \quad 0], \quad D = 0$$

Sliding Surface Design

The state trajectories must reach the sliding surface for the system to attain equilibrium [24]. The system's sliding surface must capture the fourth–order behavior required for tracking superheat dynamics. It is defined as:

$$s(t) = c_1 e_1(t) + c_2 e_2(t) + c_3 e_3(t) + e_4(t)$$

Here, $c_1, c_2, c_3 \in \mathbb{R}$ are design constants that determine the convergence dynamics and relative weighting of the error terms. The tuning of these parameters influences the responsiveness of the controller and ensures sliding stability.

Control Law Design in Sliding Mode Control

Let

$$x(t) = [y(t) \quad \dot{y}(t) \quad \ddot{y}(t) \quad \dddot{y}(t)]^T, \quad x_d(t) = [y_d(t) \quad \dot{y}_d(t) \quad \ddot{y}_d(t) \quad \dddot{y}_d(t)]^T$$

represent the system states and the desired trajectory, respectively.

The tracking error is then defined as:

$$e(t) = x(t) - x_d(t)$$

The objective of SMC is to ensure that the sliding surface $s(t) \rightarrow 0$ and to maintain the system trajectories at or near this condition. In order for the states to converge to the equilibrium defined by the sliding surface, a control law must be designed that ensures the state trajectories are driven towards and maintained on this surface, thereby effectively regulating the superheat dynamics [24].

During the convergence phase, system states are sensitive to disturbances and uncertainties. Therefore, the reaching phase should be as short as possible [24].

The SMC control law can be divided into two components:

1. Equivalent Control
2. Switching Control

Equivalent Control. Under ideal system conditions, the control input that drives the system trajectories along the sliding surface is called the *equivalent control*, $u_{eq}(t)$ [25]. The sliding surface is defined as:

$$s(t) = c_1 e_1(t) + c_2 e_2(t) + c_3 e_3(t) + e_4(t) = 0$$

To maintain the system on the sliding surface, the derivative is set to zero:

$$\dot{s}(t) = 0$$

Solving for the equivalent control yields:

$$u_{eq}(t) = -\frac{1}{B_4} \left(c_1 x_2 + c_2 x_3 + c_3 x_4 + A_{4,:} x \right)$$

where

- $A_{4,:}$: represents the last row of the state matrix A
- B_4 is the last element of input matrix B .

Switching Control: The switching control component ensures that the system remains on the sliding surface despite uncertainties and external disturbances [**nonlinear_control_book**]. The reaching dynamics are governed by the switching law.

Conventional SMC uses the discontinuous $\text{sign}(s)$ function, which can lead to high-frequency oscillations known as *chattering* [25]. To mitigate this, the signum function is replaced with a smooth hyperbolic tangent function [25]:

$$u_{sw}(t) = -K(t) \cdot \tanh\left(\frac{s}{\phi}\right)$$

where ϕ is the boundary layer coefficient that defines the thickness of the boundary layer around the sliding surface [**matlab_reachinglaw**].

The adaptive gain $K(t)$ improves convergence behavior where a higher $K(t)$ accelerates convergence but may cause larger steady-state error. On the other hand, a lower $K(t)$ reduces oscillations but results in slower convergence. Therefore, there is a need to dynamically switch between high gain and low gain depending upon the error.

Total Control Law. The complete SMC law is expressed as:

$$u(t) = u_{eq}(t) + u_{sw}(t)$$

This combined control ensures both robustness against disturbances and convergence of the system trajectories to the sliding surface equilibrium which is shown in the Results section.

4.3.2 Adaptive Fuzzy Sliding Model Control

Conventional sliding mode control (*SMC*) relies on a fixed control gain combined with a boundary layer. In the Control theory, (*SMC*) is known for its robustness, the controller may frequently results in a fast switching of the control signal known as chattering, which may cause the expansion valve to open and close rapidly in refrigeration systems. This may affect the overall efficiency of the system and increase actuator wear [25]. In order to respond to the time-varying and nonlinear dynamics of the evaporator, an adaptive *SMC* framework that can modify its control action is been proposed.

On designing the Adaptive Fuzzy Sliding Model Control *AFSMC* there are two steps which are followed:

- Designing the Adaptive Fuzzy method
- Designing the Sliding Mode Control *SMC*

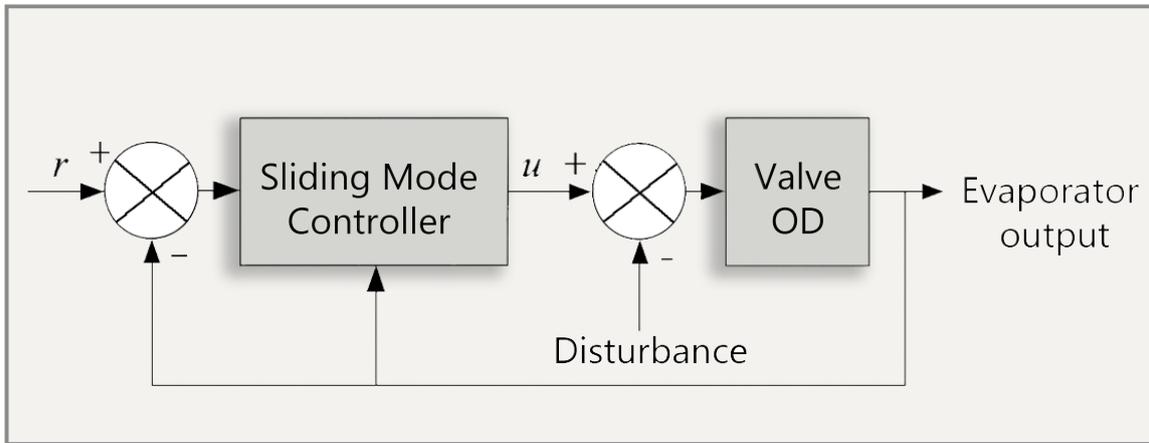


Figure 4.15: Sliding Mode Control

The representation of Adaptive Fuzzy SMC:

The valve opening degree is a combination of these two methods. Combining both design strategies achieves the *AFSMC*. As per the fuzzy design architecture, the inputs are declared as temperature superheat tracking error e , rate of change of error \dot{e} and e_{int} which prevents actuator saturation, This e_{int} is known as the integral error.

e_{int} is bounded between

$$e_{int} \in [-5, 5]$$

The goal is to reduce e to 0 while also preventing valve chattering and complying with actuator limits. The control response is varying for the non linear evaporator dynamics where a fixed parameter may fail to balance the response, the implementation of an adaptive sliding surface, whose coefficients dynamically change with the magnitude of error terms, helps mitigate these issues.

The Siding Surface $s(t)$ is defined as:

$$s(t) = \lambda_0(t) e_{int}(t) + \lambda_1(t) e(t) + \lambda_2(t) \dot{e}(t) \quad (4.30)$$

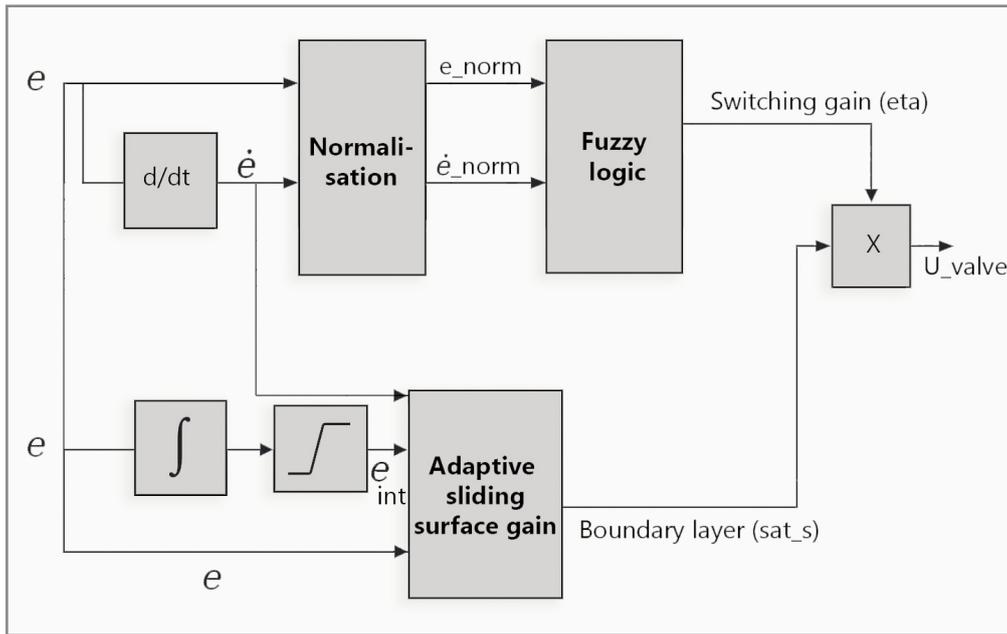


Figure 4.16: Block Diagram of Adaptive Fuzzy SMC

Here, the values of λ are the weighing coefficients which are obtained through trial and error tuning method.

- $\lambda_0(t)$ Integral coefficient
- $\lambda_1(t)$ Proportional coefficient
- $\lambda_2(t)$ Derivative coefficient

$$\lambda_0(t) = 5 + 3 |e_{\text{int}}(t)| \quad (4.31)$$

$$\lambda_1(t) = 4 + 2 |e(t)| \quad (4.32)$$

$$\lambda_2(t) = 0.5 \quad (4.33)$$

The sliding surface is designed in such a way as to make the control law more responsive by altering the impact of each error component in real time [26]. The weighting factors $\lambda_0(t)$ and $\lambda_1(t)$ are defined as increasing functions of the absolute values of the integral and proportional errors, respectively. These adaptive gains increase directly proportional to how far the system moves away from the set point, whether due to a large steady-state offset or a rapid transient change. Because of this, the sliding surface $s(t)$

becomes more sensitive to ongoing deviations, prompting the control law to take greater corrective actions to get the system back on track. The weights $\lambda_0(t)$ and $\lambda_1(t)$ decreases when the tracking error and integral error are small, indicating that system is close to the reference. The controller responds to more gently as a result, which prevents it from making excessive adjustments to the valve while the system remains stable [26].

Normalizing the input signals to the fuzzy logic controller is necessary for fuzzy inference to function within a limited computational domain. Since the fuzzy rule base is specified across a finite universe of discourse, often within the interval $[-1, 1]$, the raw inputs must be properly scaled [26].

$$e_{\text{norm}} = \max(\min(e, 1), -1) \quad (4.34)$$

$$\dot{e}_{\text{norm}} = \max(\min(\dot{e}, 1), -1) \quad (4.35)$$

The inputs of the fuzzy system stays within the specific range that aligns with the defined membership function.

Each membership function is implemented as a triangular function:

$$\mu_{\text{term}}(x) = \begin{cases} 0, & x \leq a \text{ or } x \geq c \\ \frac{x-a}{b-a}, & a < x \leq b \\ \frac{c-x}{c-b}, & b < x < c \end{cases} \quad (4.36)$$

The rules are made up of 3X3, there triangular membership function are represented for error e

Linguistic Term	Label	Triangular Range $[a, b, c]$
Negative	N	$[-1.5, -1.0, 0.0]$
Zero	Z	$[-0.5, 0.0, 0.5]$
Positive	P	$[0.0, 1.0, 1.5]$

Table 4.6: Triangular Membership Function Parameters

This rule table represents the membership functions for \dot{e}

Linguistic Term	Label	Triangular Range $[a, b, c]$
Negative	N	$[-1.5, -1.0, 0.0]$
Zero	Z	$[-0.5, 0.0, 0.5]$
Positive	P	$[0.0, 1.0, 1.5]$

Table 4.7: Triangular Membership Function Parameters

Rule No.	IF Error e is...	AND \dot{e} is...	THEN Switching Gain η is...
1	Negative (N)	Negative (N)	High (1.00)
2	Negative (N)	Zero (Z)	Medium–High (0.75)
3	Negative (N)	Positive (P)	Low (0.50)
4	Zero (Z)	Negative (N)	Medium–High (0.75)
5	Zero (Z)	Zero (Z)	Medium (0.50)
6	Zero (Z)	Positive (P)	Medium–High (0.75)
7	Positive (P)	Negative (N)	Low (0.50)
8	Positive (P)	Zero (Z)	Medium–High (0.75)
9	Positive (P)	Positive (P)	High (1.00)

Table 4.8: Fuzzy Rules for Switching Gain η

The rules are portrayed in the following format:

With weights $w_{ij} = \mu_e(i) * \mu_{\dot{e}}(j)$, The weights w_{ij} is used to calculate the numerator and the denominator to subsequently find the Valve OD gain. When the denominator is less than 0, the valve OD is limited to a value of 0.35. Contrary, when the denominator is greater than 0 the gain of Valve OD is calculated to be in the range of 0.35 and 1.

$$\eta = \max \left(\frac{\sum_{i,j} w_{ij} \eta_{ij}}{\sum_{i,j} w_{ij}}, \eta_{\min} \right), \quad \eta_{\min} = 0.35 \quad (4.37)$$

In a conventional SMC the boundary layer is fixed, whereas in case of AFSMC the a proposal of adaptive boundary is discussed below:

$$\phi(|s|) = 0.02 + 0.06e^{-|s|}, \quad \text{sat}(s) = \frac{s}{|s| + \phi(|s|)}. \quad (4.38)$$

Here, $\phi(|s|)$ is defined adaptive boundary layer which dictates the frequent switching. The less the value of $\phi(|s|)$ results in frequent switching which in turn leads to accelerated wear and tear of the valve. In contrary, when the value of $\phi(|s|)$ is larger, the frequency of the switching is reduced whereas the system doesn't perform as expected. Therefore, the value of $\phi(|s|)$ plays a crucial role in finding the balance between the system performance and valve operation.

Similarly, in case of $\text{sat}(s)$, there is a requirement to normalize the boundary layer before combining it with the adaptive fuzzy gains.

The Valve OD signal is a combination of both adaptive fuzzy gains η and normalized boundary layer $\text{sat}(s)$. Thereby, the equation combines both adaptive fuzzy logic and adaptive SMC to work in unison as one master control.

Lyapunov Stability

The stability of the proposed (*AFSMC*) system is demonstrated through a Lyapunov-based analysis. The sliding surface $s(t)$ is defined to select a standard candidate Lyapunov function [26]. Finding the appropriate energy function is one of the difficult parts. In this scenario, the most widely used energy function is used [25]. The common quadratic function used in *SMC*;

$$V(s) = \frac{1}{2}s^2 \quad (4.39)$$

To show that the closed-loop system is asymptotically stable, it is sufficient to show that the time derivative of the Lyapunov function, $\dot{V}(s)$, is negative definite for all $s \neq 0$ [25].

Taking the time derivative gives:

$$\dot{V}(s) = s \dot{s} \quad (4.40)$$

The control law formulated acts on the system to drive the s to 0. Under specific circumstances, the control term dominates the evolution of s due to the matched system dynamics, where the control input impacts the sliding variable. Now, the derivative of the Lyapunov function can be expressed as:

$$\dot{V}(s) = s \cdot \dot{s} = -\eta \cdot \frac{s^2}{|s| + \phi(|s|)} \quad (4.41)$$

Since $\eta > 0$ and $\phi(|s|) > 0$ for all s , then $\dot{V}(s) < 0$.

When system energy decreases over the period of time, the states move towards the sliding surface which is $s = 0$.

This ensures that the controller guides the superheat temperature towards the reference [25]. Together, the boundary layer function $\phi(|s|)$ and the adaptive gain selected via fuzzy logic guarantee that $\dot{V}(S) < 0$. Thus, while reducing chattering and maintaining resilience to model errors and disturbances, the *AFSMC* offers asymptotic Lyapunov stability.

Chapter 5

Simulation and Results

5.1 Results Using Transfer Function

The simulation results were obtained by taking fourth order transfer function which comprises of system dynamics of expansion valve, Evaporator and temperature superheat.

There were two stages to the validation of the suggested controllers' performance:

- Reference Tracking Verification - to make sure that any control strategy can reliably follow the reference trajectory without having to adjust.

Control Evaluation: to see how much better tracking accuracy, disturbance rejection, and robustness got once the adaptive part was included.

5.2 Reference Tracking Performance

The controllers such as *PID*, *fuzzy-PID* and *SMC* controllers were implemented to the fourth order system transfer function, from the results it is evident that the given controllers were able to follow the reference. *SMC* seems performing well when compared to other controllers but when checking in close proximity there were chattering to be found. The *MRAC* controllers follows the same trajectory as *PID* because the gains of the *PID* and *MRAC* were same. The *fuzzy-PID* follows the reference with minimal overshoot, but the concern the rise is slower when compared to other controllers.

5.2.1 Adaptive *SMC* control

The adaptive *SMC* control shows faster convergence with respect to the reference. Also, when a disturbance occurs, the adaptive *SMC* controller mitigates the disturbance and adapts to the temperature superheat reference. One disadvantage of the adaptive *SMC*

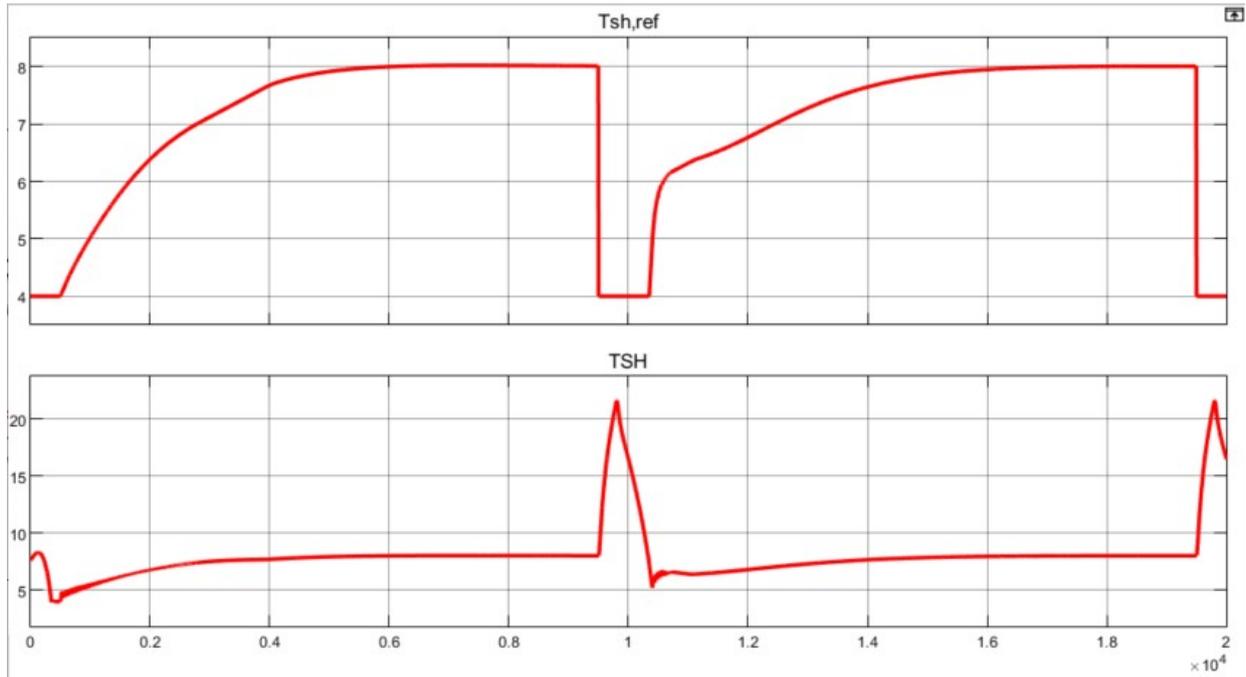


Figure 5.1: Adaptive SMC control of Temperature Superheat

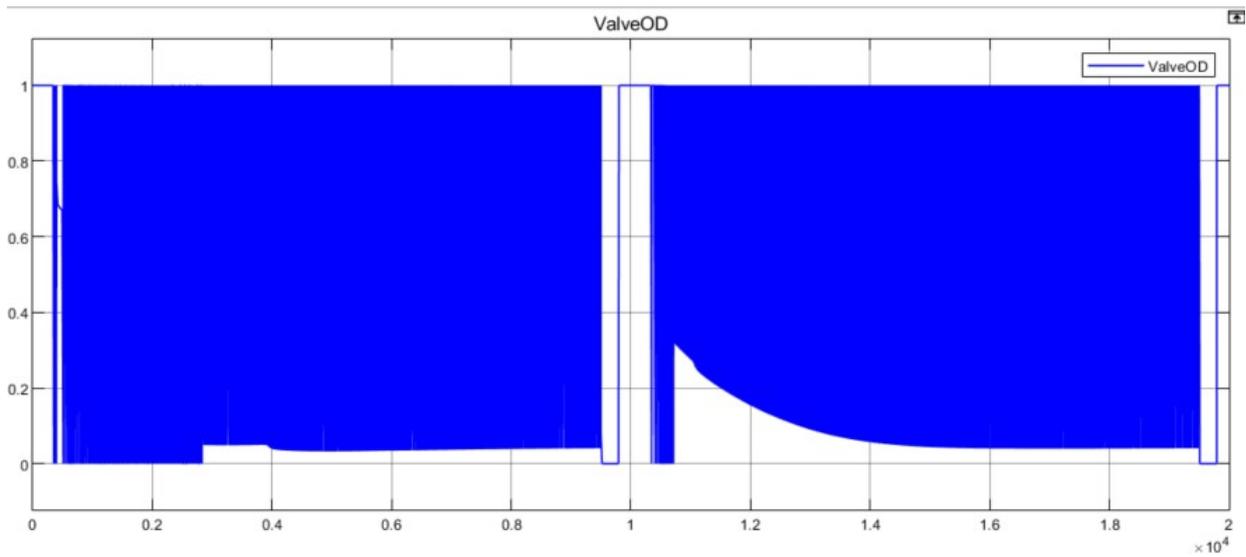


Figure 5.2: Adaptive SMC control Valve Opening Degree

controller is its inability to mitigate chattering effects in the valve OD.

5.2.2 Adaptive Fuzzy *PID*

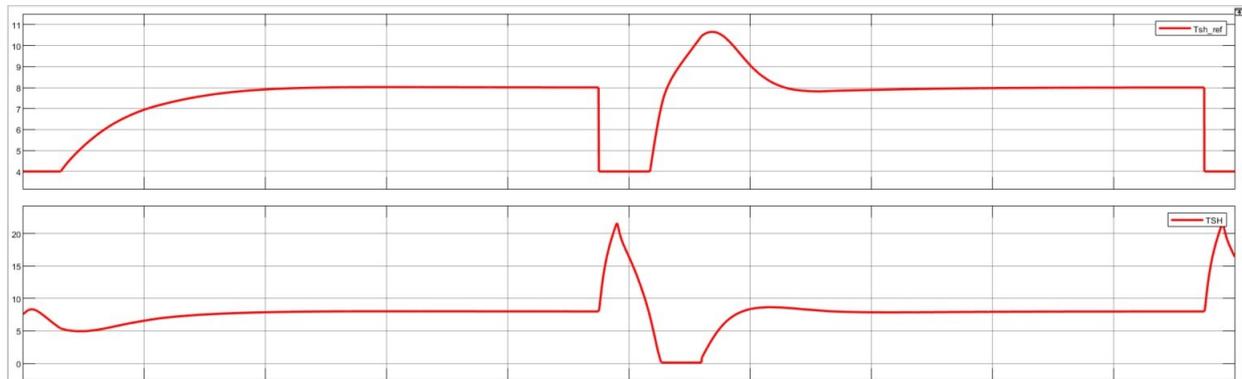


Figure 5.3: Adaptive Fuzzy *PID* of Temperature Superheat

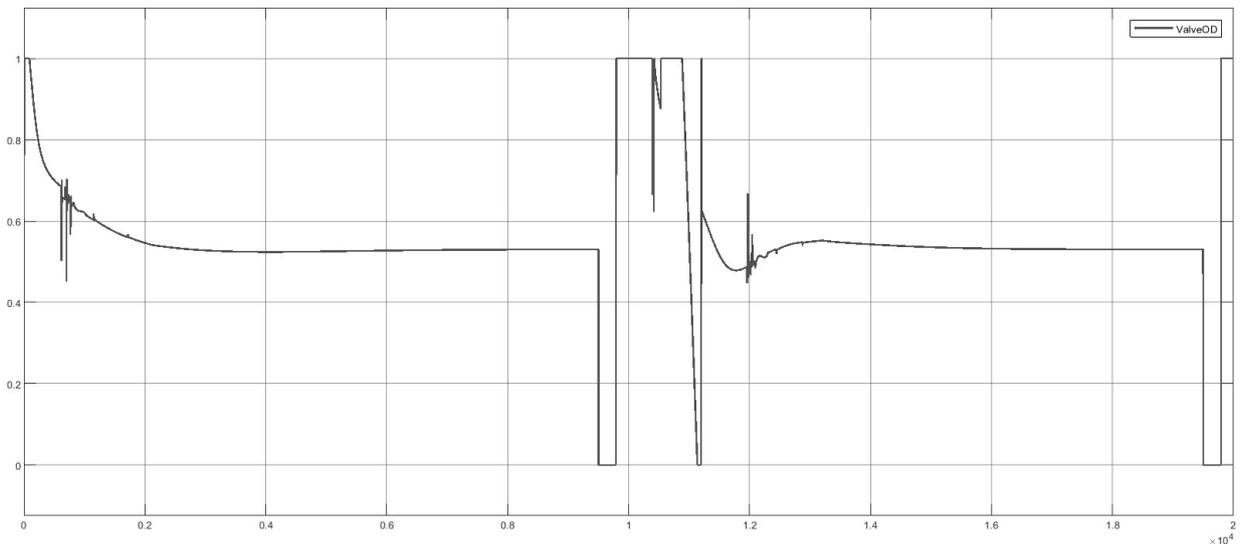


Figure 5.4: Adaptive Fuzzy *PID* of the Valve OD

The Adaptive Fuzzy *PID* control provides smooth trajectory but when noticing keenly the overtime is is higher which shouldn't be in the case of an Adaptive Fuzzy *PID*. The valve Opening degree consists of Chattering effects generating not a smooth actuation.

5.3 Adaptive Fuzzy *SMC*

The Adaptive Fuzzy *SMC* ensures a faster convergence and by nature it is a robust controller which can be seen from the above results. The chattering is completely

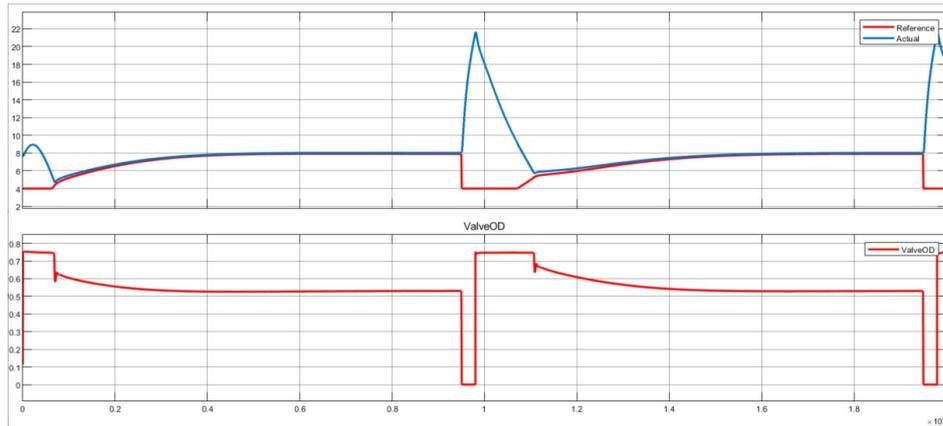


Figure 5.5: Adaptive Fuzzy SMC of TSH and the Valve OD

mitigated in the Adaptive Fuzzy SMC controller where as other adaptive controllers found to have either heavy chattering or minimal chattering.

5.4 Total Simulation

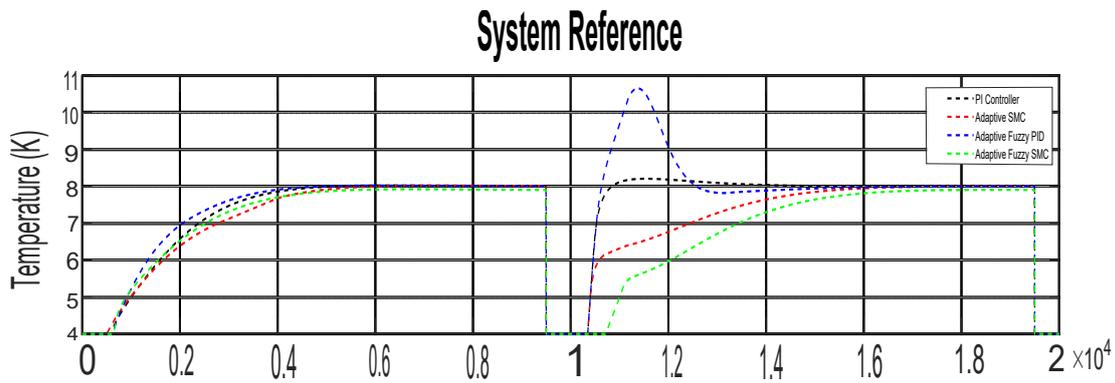


Figure 5.6: All Controllers System Reference

5.5 Adaptive Fuzzy PID with different rules

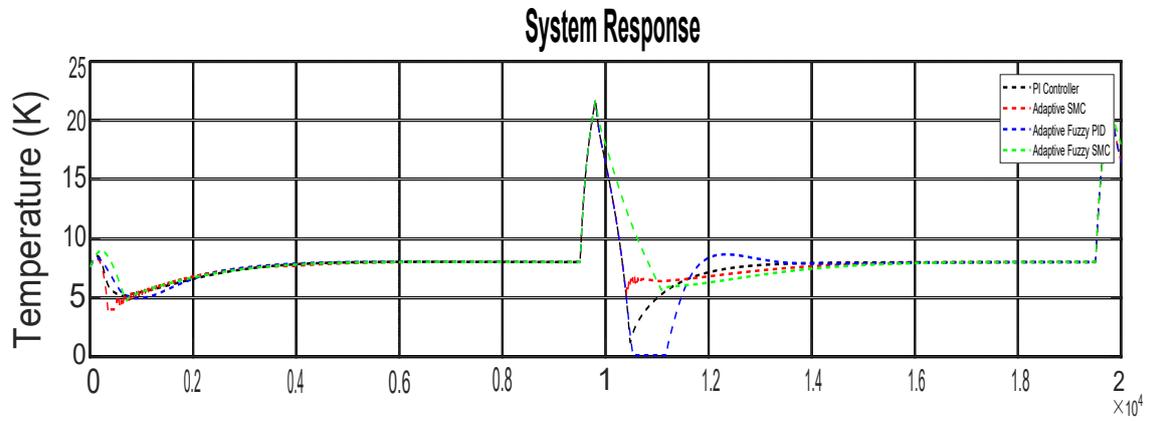


Figure 5.7: All Controllers System Response

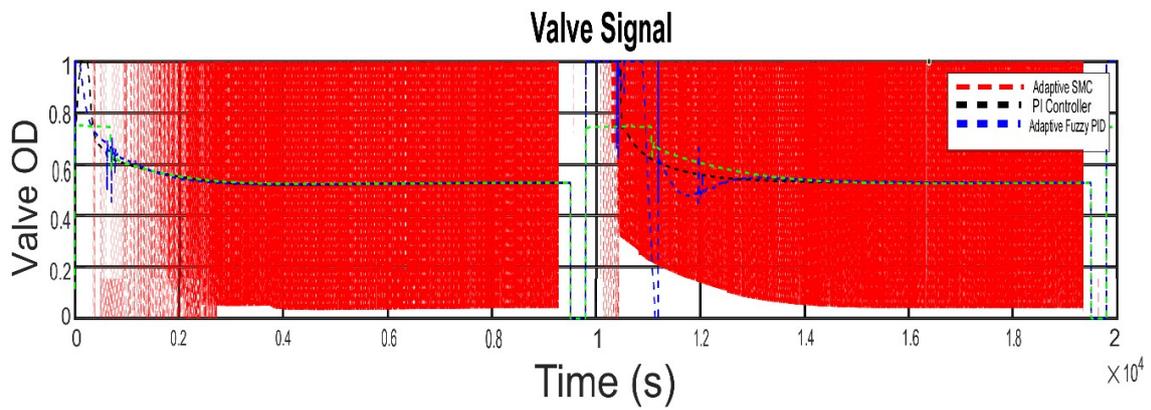


Figure 5.8: All Controllers Valve Actuation

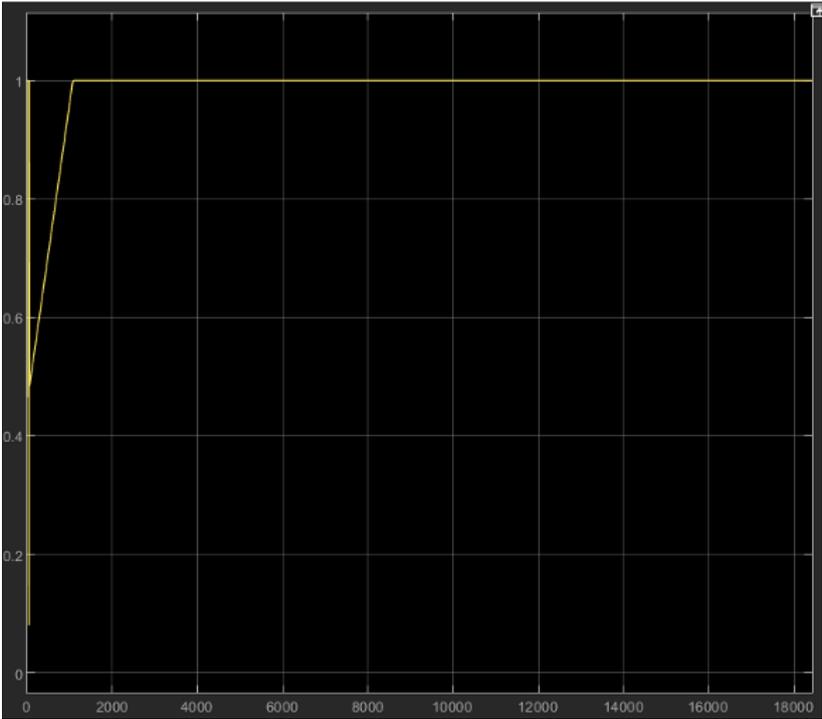


Figure 5.9: Valve Opening Degree

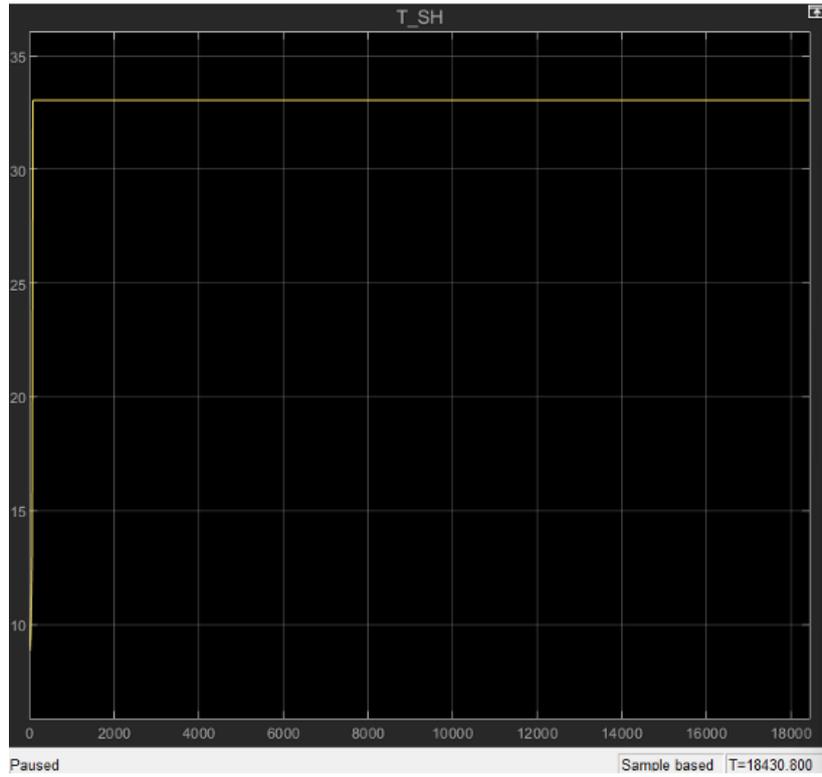


Figure 5.10: Superheat Temperature

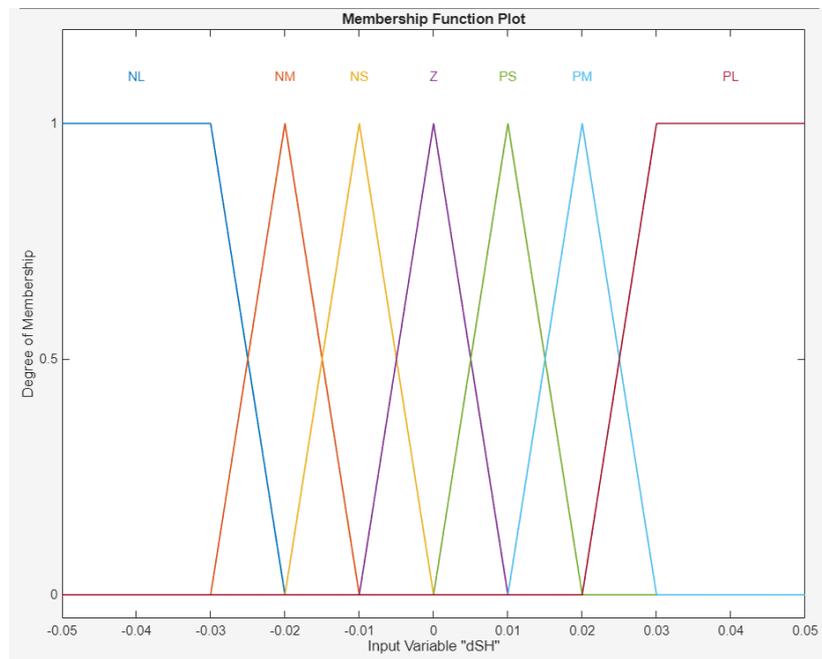


Figure 5.11: Universe of discourse for rate of change of superheat error

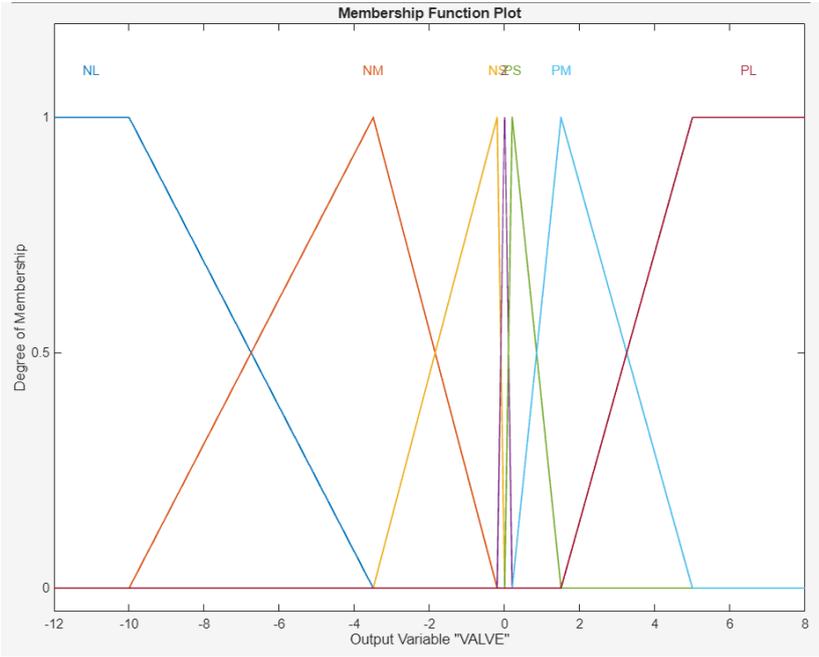


Figure 5.12: Universe of discourse for Valve

Chapter 6

Conclusion

The thesis concludes that each and every controller has its own merits and demerits. Among the control candidates, the Adaptive Fuzzy SMC found to be promising and offering the possibilities of adaptiveness with a balance of adaptability, handling disturbances without noticeable overshoot and with a Valve with no chattering thus providing a smooth actuation which prevents the valve from actual wear and tear. The traditional PI settled quick whereas fuzzy PID settled more slowly and were more sensitive to changes in the operating point, while conventional SMC reached the setpoint rapidly but at the expense of increased switching activity. The simulations are supported by the Lyapunov analysis, which shows that the selected Lyapunov candidate converges to zero and declines monotonically, suggesting asymptotic stability of the closed loop in the tested operating region and $s(t) \rightarrow 0$. In practice, this translates to less actuator wear and more stable thermodynamic functioning through stricter superheat control and softer valve action. When combined, these results support the choice of Adaptive Fuzzy SMC as the superheat loop's preferred controller.

6.1 Future Work

The control implementation can be carried out further by using Reinforcement learning, trying out with different Fuzzy Rules. Also In order to find the efficiency of the temperature superheat a compressor can be modeled so the *COP* efficiency of the superheat can be found.

Bibliography

- [1] Hiromu Yasuda, Tetsuji Yanagisawa and Minetoshi Izushi. 'A Dynamic Model of a Vapor Compression Refrigeration Cycle'. In: *Transactions of the Japan Society of Refrigerating and Air Conditioning Engineers* 11 (Sept. 2011), p. 1. DOI: 10.11322/tjsrae.11.263.
- [2] P. K. Chia et al. 'Fuzzy Control of Superheat in Container Refrigeration using an Electronic Expansion Valve'. In: *HVAC&R Research* 3.1 (1997), pp. 81–98. DOI: 10.1080/10789669.1997.10391363. eprint: <https://doi.org/10.1080/10789669.1997.10391363>. URL: <https://doi.org/10.1080/10789669.1997.10391363>.
- [3] Henrik Rasmussen, Claus Thybo and Lars Finn Sloth Larsen. 'Nonlinear superheat and evaporation temperature control of a refrigeration plant'. In: *IFAC Proceedings Volumes* 39 (19 2006), pp. 251–252. DOI: 10.3182/20061002-4-BG-4905.00043. URL: <https://www.sciencedirect.com/science/article/pii/S1474667015311757>.
- [4] Antonio Maia et al. 'Superheating control using an adaptive PID controller'. In: *HVACR Research* 20 (May 2014). DOI: 10.1080/10789669.2013.874842.
- [5] Kasper Vinther et al. 'Evaporator superheat control With one temperature sensor using qualitative system knowledge'. In: *2012 American Control Conference (ACC)*. 2012, pp. 374–375. DOI: 10.1109/ACC.2012.6314617.
- [6] H. Rasmussen, C. Thybo and L.F.S. Larsen. 'NONLINEAR SUPERHEAT AND EVAPORATION TEMPERATURE CONTROL OF A REFRIGERATION PLANT'. In: *IFAC Proceedings Volumes* 39.19 (2006). 1st IFAC Workshop on Energy Saving Control in Plants and Buildings, pp. 251–254. ISSN: 1474-6670. DOI: <https://doi.org/10.3182/20061002-4-BG-4905.00043>. URL: <https://www.sciencedirect.com/science/article/pii/S1474667015311757>.
- [7] Ivik Niels Riishøj, Christian Andersen, Rasmus Vedel Nonboe Kobborg. 'System identification of an evaporator using Gaussian Process Modeling'. In: (2020), pp. 1–98.
- [8] Zahra Sardoueinassab, Mohammad Javad Morshed and Afef Fekih. 'An Integral Terminal Sliding Mode-Based Approach to Control the Super Heat Temperature of a HVAC System'. In: *2019 American Control Conference (ACC)*. 2019, pp. 5713–5718. DOI: 10.23919/ACC.2019.8814974.

- [9] Abdelkader Outtagarts, Phillipe Haberschill and Monique Lallemand. 'The transient response of an evaporator fed through an electronic expansion valve'. In: *International Journal of Energy Research* 21 (1997), pp. 793–795. URL: <https://api.semanticscholar.org/CorpusID:97055447>.
- [10] Kartik Sharma and Dheeraj Kumar Palwalia. 'A modified PID control with adaptive fuzzy controller applied to DC motor'. In: *2017 International Conference on Information, Communication, Instrumentation and Control (ICICIC)*. 2017, p. 2. DOI: 10.1109/ICOMICON.2017.8279151.
- [11] Ying "Bai and Dali" Wang. "'Fundamentals of Fuzzy Logic Control — Fuzzy Sets, Fuzzy Rules and Defuzzifications'". In: *Advanced Fuzzy Logic Technologies in Industrial Applications*". Ed. by Ying "Bai, Hanqi Zhuang and Dali" Wang. "London": "Springer London", "2006", "17–36". ISBN: "978-1-84628-469-4". DOI: "10.1007/978-1-84628-469-4_2". URL: https://doi.org/10.1007/978-1-84628-469-4_2.
- [12] S. N. Deepa S. N. Sivanandam S. Sumathi. *Introduction to Fuzzy Logic Using MATLAB*. Springer, 2007. ISBN: 103-540-35780-7.
- [13] Hooi Tang and Nur Ahmad. 'Fuzzy logic approach for controlling uncertain and nonlinear systems: a comprehensive review of applications and advances'. In: *Systems Science Control Engineering* 12 (Aug. 2024). DOI: 10.1080/21642583.2024.2394429.
- [14] Pankaj Swarnkar and Harsh Goud. 'Design and implementation of a robust fuzzy adaptive PI controller for inherently unstable systems'. In: *International Journal of Information Technology* 17 (Oct. 2024). DOI: 10.1007/s41870-024-02239-5.
- [15] Tejash Chaudhari, Krupal Patel and Vimal Patel. 'A study of generalized bell-shaped membership function on Mamdani fuzzy inference system for Students Performance Evaluation'. In: *World Journal of Advanced Research and Reviews* 3 (Aug. 2019), pp. 083–090. DOI: 10.30574/wjarr.2019.3.2.0046.
- [16] Yanhua Bai and Dali Wang. 'Fundamentals of Fuzzy Logic Control Fuzzy Sets, Fuzzy Rules and Defuzzifications'. In: Jan. 2007, pp. 17–36. ISBN: 978-1-84628-468-7. DOI: 10.1007/978-1-84628-469-4_2.
- [17] Joe Y.M. Cheung and A. Sooban Kamal. 'Fuzzy Logic Controller for Industrial Refrigeration Systems'. In: *IFAC Proceedings Volumes* 30.6 (1997). IFAC Conference on Control of Industrial Systems "Control for the Future of the Youth", Belfort, France, 20-22 May, pp. 745–750. ISSN: 1474-6670. DOI: [https://doi.org/10.1016/S1474-6670\(17\)43454-9](https://doi.org/10.1016/S1474-6670(17)43454-9). URL: <https://www.sciencedirect.com/science/article/pii/S1474667017434549>.
- [18] Jian-Xin Xu, Chang-Chieh Hang and Chen Liu. 'Parallel structure and tuning of a fuzzy PID controller'. In: *Automatica* 36.5 (2000), pp. 673–684. ISSN: 0005-1098. DOI: [https://doi.org/10.1016/S0005-1098\(99\)00192-2](https://doi.org/10.1016/S0005-1098(99)00192-2). URL: <https://www.sciencedirect.com/science/article/pii/S0005109899001922>.

- [19] "MATLAB Help Center". "Implement Fuzzy PID Controller in Simulink". URL: <https://se.mathworks.com/help/fuzzy/implement-fuzzy-pid-controller-in-simulink-using-lookup-table.html>. "(accessed: 22.08.2025)".
- [20] Wei Jiang and Xuchu Jiang. 'Design of an Intelligent Temperature Control System Based on the Fuzzy Self-Tuning PID'. In: *Procedia Engineering* 43 (2012). International Symposium on Safety Science and Engineering in China, 2012, pp. 307–311. ISSN: 1877-7058. DOI: <https://doi.org/10.1016/j.proeng.2012.08.053>. URL: <https://www.sciencedirect.com/science/article/pii/S1877705812030640>.
- [21] Antônio A. T. Maia et al. 'Superheating control using an adaptive PID controller'. In: *HVAC&R Research* 20.4 (2014), pp. 424–434. DOI: 10.1080/10789669.2013.874842. eprint: <https://doi.org/10.1080/10789669.2013.874842>. URL: <https://doi.org/10.1080/10789669.2013.874842>.
- [22] Yongjuan Zhao and Yutitan Pan. 'The Design and Simulation of Fuzzy PID Controller'. In: *2010 International Forum on Information Technology and Applications*. Vol. 3. 2010, pp. 95–98. DOI: 10.1109/IFITA.2010.245.
- [23] Zhen-Yu Zhao, M. Tomizuka and S. Isaka. 'Fuzzy gain scheduling of PID controllers'. In: *IEEE Transactions on Systems, Man, and Cybernetics* 23.5 (1993), pp. 1392–1398. DOI: 10.1109/21.260670.
- [24] Shyam Akashe Milan Tuba. *Information and Communication Technology for Sustainable Development*. Advances in Intelligent Systems and Computing. Springer, 2018. ISBN: 978-981-13-7166-0.
- [25] JEAN-JACQUES E. SLOTINE. *APPLIED NONLINEAR CONTROL*. Nonlinear control theory. Prentice Hall, 1991. ISBN: D-13-DHDfiTa-S.
- [26] Byungkook Yoo and Woonchul Ham. 'Adaptive fuzzy sliding mode control of nonlinear system'. In: *IEEE Transactions on Fuzzy Systems* 6.2 (1998), pp. 315–321. DOI: 10.1109/91.669032.