PEOPLE'S DEMOCRATIC REPUBLIC OF ALGERIA
MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
UNIVERSITY 20 AOUT 1955 – SKIKDA
FACULTY OF TECHNOLOGY
DEPARTEMENT OF PETROCHIMICAL AND PROCESS ENGINEERING

Memory

For the degree of

Master

Section: Petrochemical industries
Specialty: Automation in Petrochemical industries

Design of an Advanced Fuzzy Controller for a Binary Distillation Column

Presented by:
GUERILLI Lamine
DJARRDIR Taki-Eddine

Supervisor:
Dr. BENDIB Riad

Year: 2025-2024

Acknowledgements

At the end of this thesis, we would like to thank God for giving us the strength, the health, and the patience to complete this work.

Our deepest gratitude goes to our supervisor, Mr. BENDIB Riad, who suggested this research topic and guided us with valuable advice and support from the beginning until the final stage.

We are also sincerely thankful to the president and members of the jury for taking the time to review and evaluate our work.

Finally, we would like to thank all the professors of the Petrochemistry department for their dedication to our training and for always being available to help and guide us.

THANK YOU

Dedicate

I dedicate this work to my dear parents, who have supported me with all their strength throughout my journey; without them, I would never get where I am.

To all my family.

To my partner Taki.

To all my dearest friends.

Finally, I dedicate this work to anyone who has contributed directly or indirectly to its realization.

Lamine

Dedicate

I would like to dedicate this work to my beloved father, whose moral and material support has always been by my side until today.

To my precious mother, no words, however sincere, could ever fully express the depth of love and affection I hold for you.

To my brothers and my sister, whom I cherish deeply, for their encouraging words and their constant support.

To my dear partner, Lamine.

To all my friends who stood by me and offered their help during challenging times.

Lastly, I wish to acknowledge myself for the determination, effort, and sacrifices I have made to complete this humble work.

taki

Abstract:

Fuzzy logic (FL) is a reasoning approach inspired by human thinking, developed to reduce the uncertainties associated with vague or imprecise information. Most fuzzy logic controllers are built using type-1 fuzzy sets and are referred to as type-1 FLCs. However, recent studies have shown that a new class of controllers, called type-2 FLCs, is more capable of managing the nonlinearities and high levels of uncertainty present in complex systems.

This brief thesis focuses on the control of three different systems using PID, type-1 fuzzy logic, and type-2 fuzzy logic. After presenting a theoretical background covering the essential concepts and definitions, we developed the modeling and control strategies for these processes, applying both type-1 and type-2 fuzzy logic as well as PID controllers.

Finally, simulation results obtained in MATLAB demonstrate the superior performance of the type-2 fuzzy controller compared to both the type-1 controller and the PID controller.

Key words: fuzzy controllers, fuzzy logic type-1, fuzzy logic type-2.

Résumé:

La logique floue (FL) est une approche de raisonnement inspirée par la pensée humaine, développée pour réduire les incertitudes associées à des informations vagues ou imprécises. La plupart des régulateurs flous sont construits à partir d'ensembles flous de type 1 et sont appelés FLC de type 1. Cependant, des études récentes ont montré qu'une nouvelle catégorie de régulateurs, appelés FLC de type 2, est plus apte à gérer les non-linéarités et les niveaux élevés d'incertitude présents dans les systèmes complexes.

Ce mémoire bref se concentre sur le contrôle de trois systèmes différents en utilisant un PID, la logique floue de type 1, et la logique floue de type 2. Après avoir présenté un cadre théorique couvrant les concepts et définitions essentiels, nous avons développé la modélisation et les stratégies de commande de ces processus, en appliquant la logique floue de type 1 et de type 2 ainsi que les régulateurs PID.

Enfin, les résultats de simulation obtenus sous MATLAB démontrent la performance supérieure du régulateur flou de type 2 par rapport au régulateur de type 1 et au régulateur PID.

Mots clés: contrôleurs flous. Logique floue de type 1, logique floue de type 2.

الملـــــخص:

المنطق الضبابي (FL) هو أسلوب في الاستدلال مستوحى من التفكير البشري، تم تطويره بهدف تقليل الشكوك المرتبطة بالمعلومات الغامضة أو غير الدقيقة. تُبنى معظم المتحكمات الضبابية باستخدام مجموعات ضبابية من النوع الأول، وتسمى متحكمات ضبابية من النوع الأول FLC) من النوع 1 .(ومع ذلك، أظهرت دراسات حديثة أن فئة جديدة من المتحكمات، وهي متحكمات ضبابية من النوع الثاني FLC) من النوع 2(، قادرة بشكل أفضل على التعامل مع اللاخطيات ومستويات عدم اليقين العالية الموجودة في الأنظمة المعقدة.

يركز هذا البحث الموجز على التحكم في ثلاثة أنظمة مختلفة باستخدام المتحكمPID ، والمنطق الضبابي من النوع الأول، والمنطق الضبابي من النوع الثاني. بعد تقديم خلفية نظرية تشمل المفاهيم والتعريفات الأساسية، قمنا بتطوير جزء النمذجة واستراتيجيات التحكم لهذه العمليات، من خلال تطبيق المنطق الضبابي من النوعين الأول والثاني، بالإضافة إلى المتحكم.PID

وأخيرًا، أظهرت نتائج المحاكاة التي تم الحصول عليها باستخدام برنامج MATLAB فعالية المتحكم الضبابي من النوع الثاني مقارنةً بالمتحكم الضبابي من النوع الأول وبالمتحكم.PID

الكلمات المفتاحية: المتحكم الضبابي، المنطق الضبابي من النوع 1، المنطق الضبابي من النوع 2.

Table of contente

List of figures:

List of tables:

List of abbreviations:

PID: Proportional-Integral- Derivative controller.

COG: centre of gravity.

COS: centre of sets.

FBF: fuzzy logic function.

FIS: Fuzzy inference systems.

FLC: Fuzzy logic control.

FLS: Fuzzy logic system.

FS: Fuzzy sets.

MeOM: Mean of maximum.

MFs: Membership functions.

MIMO: Multiple input multiple outpout.

MISO: Multiple input single outpout.

N/NN: Negative.

P/PP: Positive.

SISO: Single input single outpout

T1FLS: Type 1 fuzzy logic system

T2FLS: Type 2 fuzzy logic system

Z/ZZ: Zero

μ: Membership degree

General introduction

General introduction:

Logic, in its common understanding, is viewed as a framework for thinking processes that must remain precise, rigorous, and formal. However, mathematicians have recognized that there is not just a single logic, but as much logic as one wishes, each depending on the chosen axiomatic system [1].

Fuzzy logic focuses on representing imprecise knowledge and enabling approximate reasoning. It can therefore be related to problem-solving heuristics, expert systems, artificial intelligence, and even natural language processing [2].

The main motivation behind introducing fuzzy set theory was the need to model real-world phenomena, which are inherently vague and ambiguous. Human knowledge about complex problems can be effectively expressed through the imprecise terms of natural language. Fuzzy set theory and fuzzy logic offer formal tools for mathematically representing and efficiently processing this type of information [3].

In today's modern world, there are high levels of uncertainty that affect operations across many industries. As a result, researchers are increasingly working to find solutions that can help reduce these uncertainties, particularly in control system applications. For this reason, classical fuzzy logic, called type-1 fuzzy logic, has been generalized to a new framework known as type-2 fuzzy logic.

It has been shown that type-1 Fuzzy Logic Controllers struggle to manage high levels of uncertainty, while the newer type-2 Fuzzy Logic Controllers demonstrate improved performance compared to both type-1 FLCs and PID controllers. The type-2 FLCs can overcome the limitations of type-1 systems and PID controllers, leading to more advanced and efficient fuzzy logic control strategies [4].

The objective of this work is to present and evaluate the control technique applied to our binary distillation column model for wood and berry.

The organization of our work is as follows:

In the first chapter, we presented an artificial intelligence technique known as fuzzy logic, focusing on type-1 fuzzy systems. We introduced all the concepts and definitions required to understand this method, along with the structural components of a type-1 fuzzy system.

Chapter II then extended these ideas to a more advanced framework called type-2 fuzzy logic. This newer approach is particularly effective in situations where it is extremely difficult to specify precise membership functions for a fuzzy system. As a result, type-2 fuzzy logic makes it possible to incorporate uncertainty directly into the rules, which in turn influences the output of the system under study.

The **third chapter** presented the description, the concept and the modeling of the systems.

The final chapter focuses on simulating and analyzing results in matlab obtained through the application of type-1 and type-2 fuzzy logic controllers, as well as PID controllers.

Lastly, this work will conclude with a general summary that brings together all the results achieved, along with recommendations and suggestions for future research in this field.

CHAPTER I: FUZZY LOGIC
TYPE 01

I-1 Introduction:

Several decades ago, probability theory was the main tool available to experts for managing uncertainty in problems related to science, technology, and daily life. Probability, which relies on the rules of traditional two-valued logic, is appropriate for dealing with uncertainty caused by randomness. However, it is not adequate when the uncertainty is related to vague or imprecise information.

This situation changed with the introduction of fuzzy set theory by Zadeh in 1965, as well as fuzzy logic derived from this theory. These new mathematical frameworks provide scientists with the means to model situations that are not clearly defined, making it possible to solve problems described in natural human language in a mathematical way.

As a result, their areas of application have grown quickly, covering not only the physical sciences but also economics, management, expert systems (such as financial planning, diagnostics, weather forecasting), information retrieval, control systems, industrial processes, robotics, decision support, computer programming, medicine, biology, the humanities, education, and almost every other aspect of human activity, including the study of human reasoning itself. [5]

I-2 Fuzzy logic:

Fuzzy concepts and fuzzy logic are used so often in our daily lives that we usually do not even notice them. For example, when responding to survey questions, people often answer with phrases like "Not Very Satisfied" or "Quite Satisfied," which are vague or unclear. But what exactly does that mean? To what extent is a person satisfied or dissatisfied with a service or product? Such vague expressions can only be created and understood by humans, not by machines.

Is it possible for a computer to answer those survey questions in the same way as a human? That is not possible, because computers only understand values like "0" or "1", or "HIGH" and "LOW". These types of data are called crisp or classical data, and they can be processed by all machines [6].

I-3 The history of fuzzy logic:

When Aristotle and earlier philosophers developed their theories of logic and mathematics, they established what is called the **Law of the Excluded Middle**, which states that every statement must be either true or false. For example, grass is either green or not green; it cannot be both at the same time. However, not all thinkers agreed with this strict view. Plato, for instance, suggested there could be a third area beyond true and false, where these opposites mix or overlap.

In Aristotle’s perspective, logic was always based on two values. Later, in the 19th century, George Boole created a mathematical system of algebra and set theory that described this two-valued logic, assigning 1 to true and 0 to false. In the early 20th century, Jan Lukasiewicz introduced a three-valued logic (true, possible, false), but this idea did not achieve wide acceptance.

In 1965, Lotfi A. Zadeh from the University of California at Berkeley published **“Fuzzy Sets”**, which laid the foundation for fuzzy set theory and fuzzy logic. Zadeh noticed that classical computer logic could not process information containing subjective or vague concepts, so he created fuzzy logic to enable computers to work with data in shades of gray, closer to how humans' reason.

Figure 1: Lotfi A Zadeh

Although this technology was first introduced in the United States, scientists and researchers in Europe generally ignored it for many years. This might have been due to its unusual name, which they found hard to take seriously, as it seemed almost childish.

Some mathematicians claimed that fuzzy logic was simply another form of probability theory. However, fuzzy logic gained quick acceptance in countries such as Japan, China, and other parts of Asia. Currently, China has the largest community of fuzzy logic researchers, with more than 10,000 scientists working in this field. Although Japan is seen as a leader in fuzzy research, it actually has fewer specialists involved. Around ten years ago, the Chinese University of Hong Kong carried out a survey of consumer products that use fuzzy logic and published a report of more than 100 pages covering washing machines, camcorders, microwave ovens, and many other electrical and electronic devices [7].

I.3.1 The first application of fuzzy logic:

This invention did not gain significant recognition until 1974, when Dr. E. H. Mamdani, a professor at London University, used fuzzy logic in a practical way to control an automatic steam engine. This was nearly a decade after fuzzy theory had first been proposed. Later, in 1976, Blue Circle Cement together with SIRA in Denmark created an industrial application to regulate cement kilns, and that system became operational in 1982.

Since the 1980s, there has been a steady rise in reports of fuzzy logic implementations, covering areas such as industrial manufacturing, automatic control, automobile production, banking, healthcare, libraries, and academic education. Today, fuzzy logic methods are broadly applied throughout society [6].

I.4 fuzzy logic characteristics:

Here are some key features of fuzzy logic [8]:

- It is highly flexible and simple to apply.
- It supports the imitation of human reasoning patterns.
- It makes it possible to design non-linear functions with any desired level of complexity.
- It is developed with full guidance from domain experts.
- In fuzzy logic, inference works by propagating flexible constraints.
- It is a very suitable approach for reasoning under uncertainty

I.5 Fuzzy logic type 01 basics:

I.5.1 fuzzy sets:

Fuzzy logic relies on the principles of fuzzy set theory, which extends the ideas of classical set theory. Describing fuzzy set theory as a generalization of classical set theory means that classical sets are actually a specific case within fuzzy sets. In other words, using set theory language, classical set theory can be viewed as a subset of fuzzy set theory.

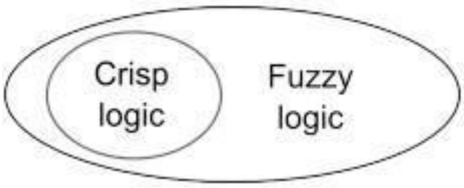


Figure 2 :The classical set theory is a subset of the theory of fuzzy sets

Fuzzy logic is built upon fuzzy set theory, which generalizes classical set theory [Zadeh, 1965]. In the literature, it is common — though somewhat imprecise — to refer to these simply as “fuzzy sets” instead of “fuzzy subsets.” Classical sets are sometimes called “crisp sets” to contrast them with fuzzy or vague sets, and likewise, classical logic is often known as Boolean or binary logic [9].

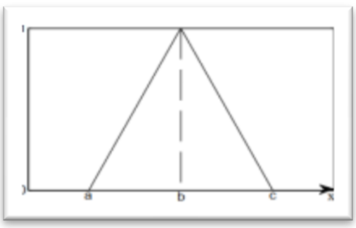
I.5.2 The membership function:

As we have established, fuzzy logic is not itself unclear or "fuzzy," but rather a system designed to describe and handle fuzziness. This fuzziness is most effectively represented through a membership function. Put simply, the membership function expresses the degree of truth within fuzzy logic.

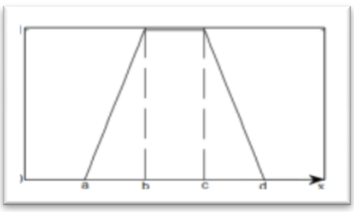
I.5.3 Membership function types:

The type of membership function used determines the type of fuzzy sets that will result. Zadeh introduced several membership functions, which can be grouped into two main categories: "linear" functions, formed by straight lines, and "nonlinear" or curved functions. However, nonlinear functions generally require more computational time. For this reason, most practical applications rely on linear membership functions. Let us now examine some examples of these membership functions [10].

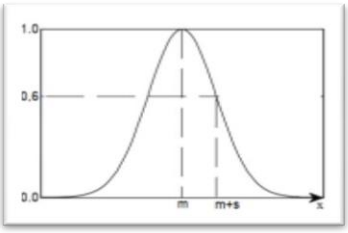
- Triangular function
- Trapezoide function
- Gaussian function
- Segmoide function



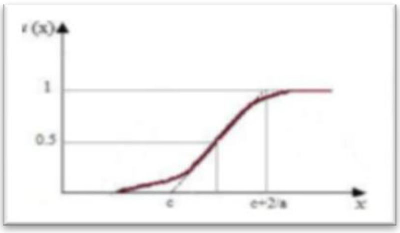
-triangular function



-trapezoid function



-Gaussian function



Sigmoid function

How to Select an Appropriate Membership Function?

Membership functions (MFs) can take any shape or form, as long as they map the data to the desired degrees of membership. The selection of a suitable MF is ultimately left to the user's decision. This flexibility is one of the strengths of fuzzy systems, as it allows for personal degrees of freedom. With experience, one can learn which shapes of membership functions work best for a particular application.

I.5.4 Features of membership function:

The core of a Membership Function: The core of a membership function for a fuzzy set AAA is defined as the region within the universe of discourse where elements have completed or full membership in AAA. In other words, the core includes all elements xxx of the universe for which: [10]

$$\mu_A(x) = 1$$

The support of a Membership Function: The support of a membership function for a fuzzy set AAA is defined as the region within the universe of discourse where the membership value is non-zero. In other words, the support includes all elements xxx in the universe for which:

$$\mu_A(\mathrm{x}) > 0$$

The boundary of a Membership Functions: The boundary of a membership function for a fuzzy set AAA refers to the region within the universe XXX where elements have a membership value that is non-zero but not equal to full membership. In other words, the boundary includes those elements xxx of the universe of discourse whose membership values satisfy:

$$\mu_A(x) \in (0,1)$$

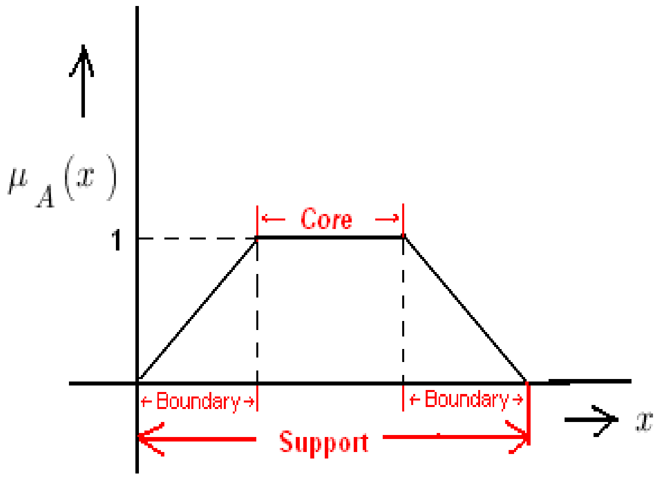


Figure 3: Core, Support and Boundary of a membership function representation.

- **An α-cut**: This represents the classical subset of elements whose membership degree is greater than or equal to α: α-cut (A) $=\{x \in X \mid \mu A(x) \geq \alpha\}$.
- **Cross-over Points of a MB:** It is defined as the collection of elements in a fuzzy set AAA that have a membership value exactly equal to 0.5.
- **Height of a MB:** The height of a membership function refers to its maximum membership value. If the height of a fuzzy set is less than 1, it is called a subnormal fuzzy set. On the other hand, if its height equals 1, it is considered a normal fuzzy set [10].

I.5.5 Fuzzy operators:

To make the handling of fuzzy sets easier, the classical set theory operators are redefined to match the membership functions used in fuzzy logic for values strictly between 0 and 1. Unlike the fixed properties of fuzzy sets, the definitions of their operators can be chosen in a similar way to membership functions. Below are the two main sets of operators most commonly applied for complement (NOT), intersection (AND), and union (OR).

Table 1 :Fuzzy operators:

Name	Intersection AND $\mu_{A\cap B}$ (x)	Union OU $\mu_{A\cup B}$ (x)	Complement NOT $\mu_{\bar{A}}$ (x)
Zadeh Operators MIN/MAX	min (μ_A(x), μ_B(x))	max (μ_A(x), μ_B(x))	$1 - \mu_A$(x)
Probabilistic PROD/PROBOR	$\mu_A(x) \times \mu_B(x)$	$\mu_A(x) + \mu_B(x) - \mu_A(x) \times \mu_B(x)$	$1 - \mu_A$(x)

Using the standard definitions of fuzzy operators, the classical properties of commutativity, distributivity, and associativity are still preserved.

I.6 linguistic variables:

The concept of the membership function, as discussed earlier, makes it possible to define fuzzy systems using natural language. The membership function links fuzzy logic with linguistic variables, which will now be introduced.

For example, let V be a variable (such as service quality or the amount of a tip), X the range of possible values for this variable, and TVa finite or infinite collection of fuzzy sets. A linguistic variable is then described by the triplet (V,X,TV).

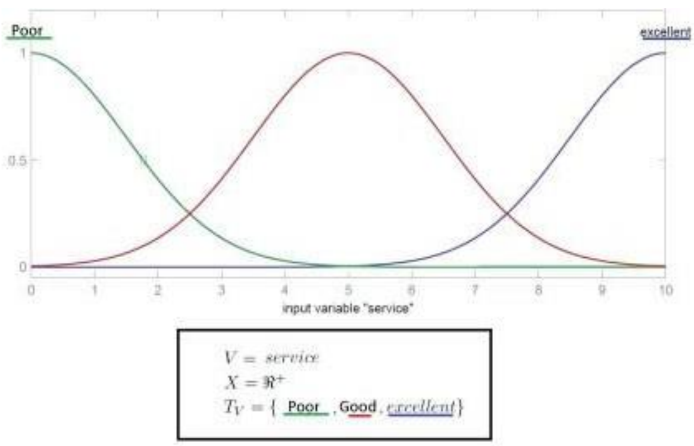


Figure 4: Linguistic variable ‘quality of service’

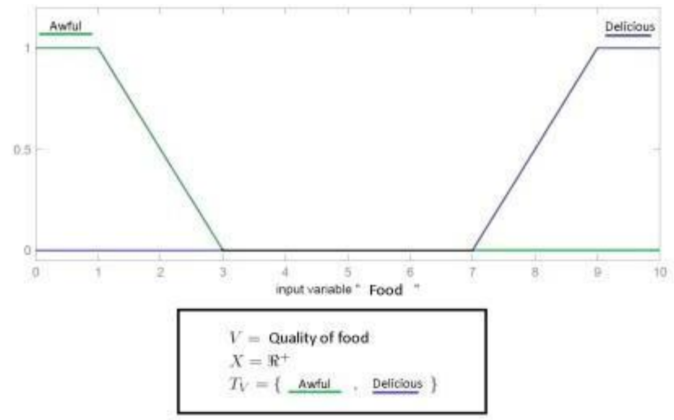


Figure 5: Linguistic variable 'quality of food'

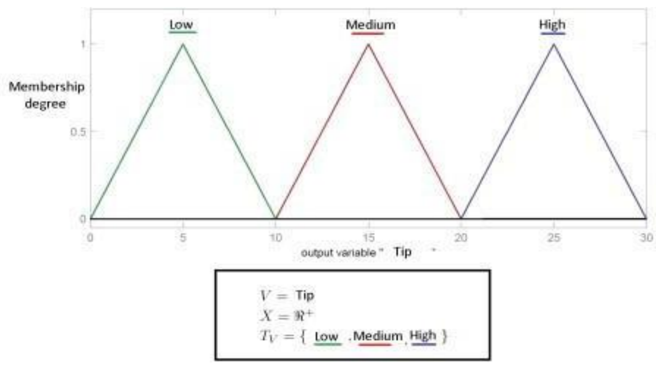


Figure 6: Linguistic variable 'tip amount'

When defining the fuzzy sets for linguistic variables, the objective is not to cover every possible description of these variables in detail. Instead, we only define a limited number of fuzzy subsets that will actually be useful later when establishing the rules to be applied. For example, we did not include a subset labeled "average" for food quality because it would not be relevant to our rule base.

In a similar way, consider that a tip of 30 is technically higher than 25, yet 25 may have a higher degree of membership in the fuzzy set "high" compared to 30. This happens because 30 could be interpreted as "very high" (or even "excessive," if you prefer), but since we did not define a fuzzy set for "very high," it was not needed in our rules [9].

I.7 fuzzy system type-1 structure:

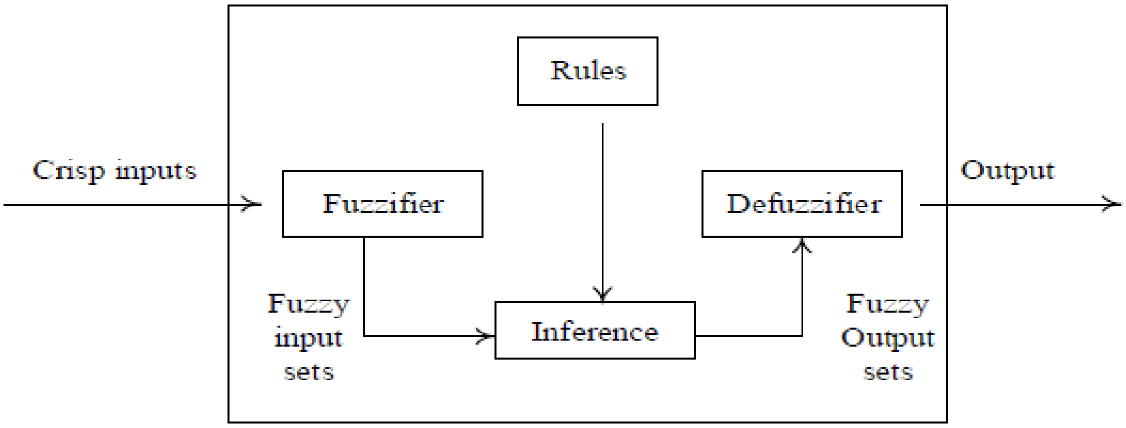


Figure 7: Type-1 fuzzy logic system diagram

I .7.1 Fuzzification:

Fuzzification refers to the process of transforming a crisp quantity into a fuzzy one. This is based on the understanding that many quantities we treat as crisp and deterministic are, in fact, subject to considerable uncertainty. When this uncertainty is caused by imprecision, vagueness, or ambiguity, the variable is best modeled as fuzzy and can be described through a membership function.

In practice, hardware such as a digital voltmeter produces crisp data, yet this data can still include measurement errors. **Figure 8** illustrates a possible error range for a typical voltage measurement, along with a membership function that might capture this imprecision.

Representing imprecise data as fuzzy sets is a helpful, though not strictly required, step when using these data within fuzzy systems. This idea is demonstrated in **Figure 9**, where we see the data as a crisp value in **Figure 9(a)** and as a fuzzy value in **Figure 9(b).**

In Figure 9(a), for example, a crisp voltage measurement might be compared to a fuzzy set labeled “low voltage.” The figure shows that the crisp measurement intersects the fuzzy set at a membership degree of 0.3, meaning the crisp value is compatible with “low voltage” to that extent. In Figure I.9(b), the intersection of the fuzzy set “medium voltage” with a fuzzified voltage measurement reaches a membership of 0.4. We can observe in **Figure 9(b)** that the intersection between these two fuzzy sets forms a small triangle, with its highest membership degree at 0.4 [11].

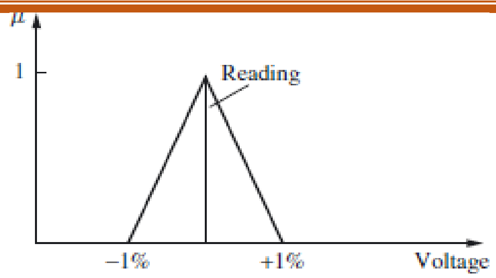


Figure 8: Membership function representing imprecision in "crisp voltage reading."

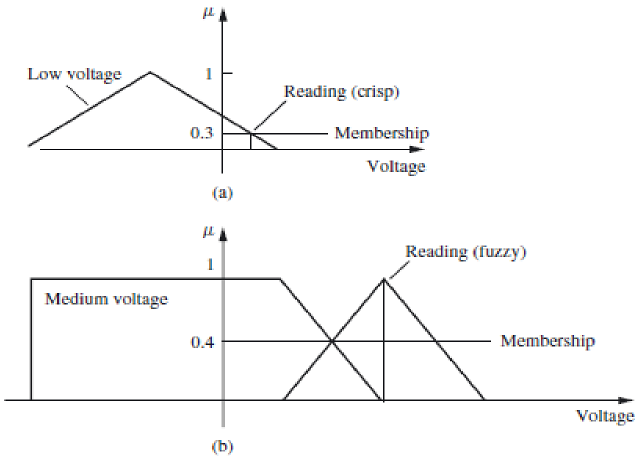


Figure 9: Comparisons of fuzzy sets and crisp or fuzzy readings(a)Fuzzy set and crisp reading; (b) Fuzzy set and fuzzy reading.

I.7.2 Fuzzy Rule Base and Inference Engine:

A fuzzy system uses a reasoning process that relies on IF...THEN rules, where each rule defines a specific result. A rule is activated when its input condition matches the IF part of the rule, leading to an output defined by the THEN part. Within a fuzzy logic system, many rules can exist, each connected to one or more IF conditions. A rule might also include several input conditions, joined by logical AND or OR relationships to trigger its outcome.

In a MISO (multiple input, single output) scenario, the system is usually described through a set of rules with the following structure (for two inputs and one output):

$$R_1\text{: If x is } A_1 \text{ , … and y is } B_1 \text{ then z is } C_1$$

$$R_2 \text{ : If x is } A_2 \text{ , … and y is } B_2 \text{ then z is } C_2$$

$$R_n \text{ : If x is } A_n \text{ , … and y is } B_n \text{ then z is } C_n$$

Where x, y and z are linguistic variables representing the FS inputs and output, respectively.

Ai, Bi and Ci are linguistic values of the variables x, y and z in the universe of discourse U, V and W, respectively with i = 1, 2, …, n.

Fuzzy inference involves combining the system inputs, the output membership functions, and the fuzzy rules to produce a fuzzy output. There are two main kinds of Fuzzy Inference Systems (FIS): the Mamdani type and the Takagi-Sugeno-Kang type.

I.7.2.1 Mamdani Fuzzy Inference Systems:

The most widely used fuzzy inference technique is the Mamdani method, originally introduced by Mamdani and Assilian. In a Mamdani-type system, the output of each rule is represented as a fuzzy set. Because Mamdani systems use rule bases that are more intuitive and easier to interpret, they are especially appropriate for expert systems where rules are derived from human expert knowledge.

The inference process for a Mamdani system is illustrated in Figure I.10.

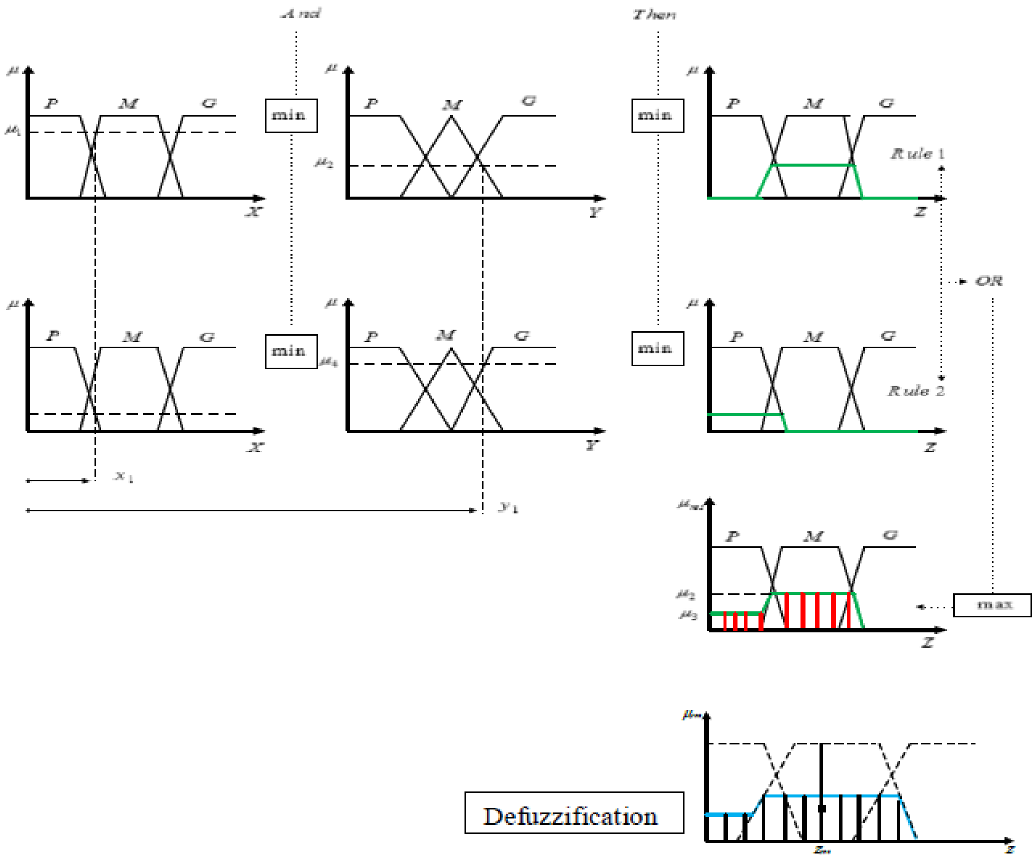


Figure 10: Mamdani's Min–Max inference method

In Mamdani's model, fuzzy implication is handled using Mamdani's minimum operator. The conjunction operator is also defined as the minimum function, while the t-norm used in the compositional rule is likewise the minimum. For aggregating the outcomes of the rules, the maximum operator is applied [12].

The output of each rule is a fuzzy set obtained from the output membership function and the implication method used by the FIS. These fuzzy output sets are then merged into a single fuzzy set through the FIS aggregation method. Finally, to produce a crisp output value, this combined fuzzy set is defuzzified using one of the available defuzzification techniques.

I.7.2.2 Takagi-Sugeno-Kang Fuzzy Inference System:

The core idea of Takagi-Sugeno-Kang fuzzy inference, often called Sugeno fuzzy inference, is that the conclusion of each rule is not expressed in symbolic terms, but instead takes a numerical form, usually as a linear combination of the inputs [12]. The defuzzification step in a Sugeno system is also more computationally efficient than in a Mamdani system, since it relies on calculating a weighted average or weighted sum of a small set of data points, instead of finding the centroid of a two-dimensional area [12].

Each rule in a Sugeno-type system works as illustrated in Figure I.11, which shows a system with two inputs, xxx and yyy. In this context, the conclusion of each rule in the collection of fuzzy IF–THEN rules are written in the following format:

$$\boldsymbol{R^{(1)}: If\ x\ is\ F_1^1\ and\ y\ is\ F_2^1\ then\ z^1 = a^1 x + b^1 y + c^1}$$

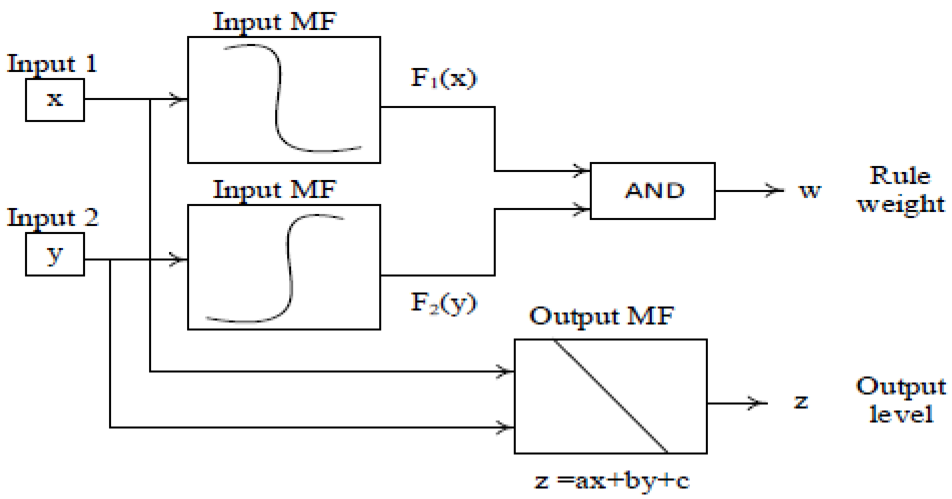


Figure 11: Diagram shows rule operations in Sugeno system

- Z^1: Rule output level, which is either a constant value or a linear function of the input values:

Here, x and y are the values of input 1 and input 2, respectively, and a^1, b^1 , $and\ c^1$ are constant coefficients. For a zero-order Sugeno system, z^1 is a constant ($a^1 = b^1 = 0$).

- w^1 : Rule firing strength derived from the rule antecedent here, F_1 and F_2 are the membership functions for inputs 1 and 2, respectively.

The output of each rule is the weighted output level, which is the product of w^1 and z^1.

The easiest way to visualize first-order Sugeno systems (a^1 and b^1 are non-zero) is to think of each rule as defining the location of a moving singleton. That is, The singleton output spikes are able to shift linearly within the output space according to the input values. The strength of the rule activation then determines the magnitude of the singleton spike [12].

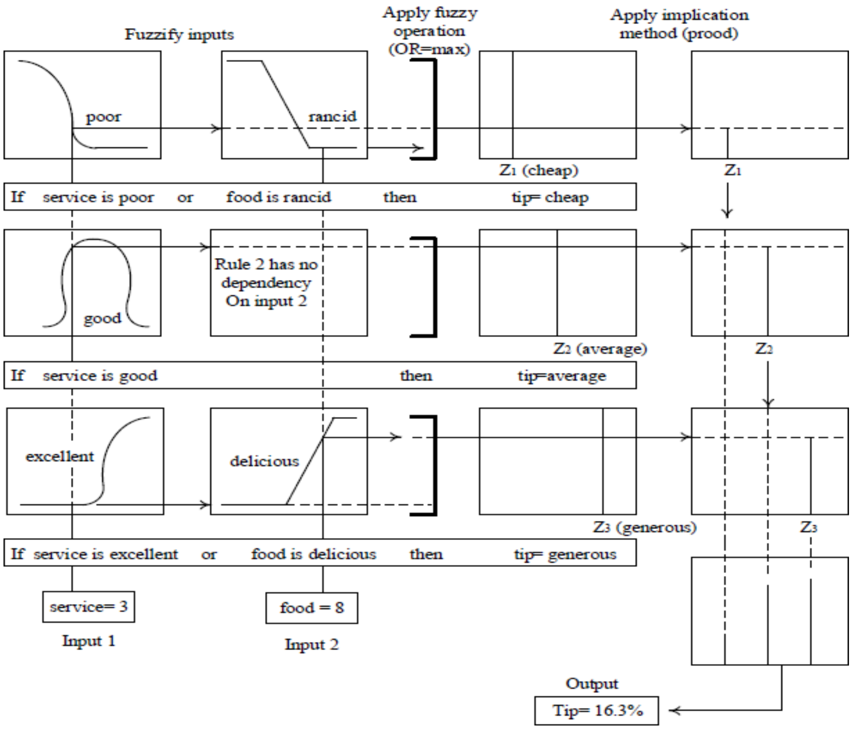


Figure 12: Fuzzy inference process for a Sugeno system

I.7.2.3 Reasoning in fuzzy logic:

In classical logic, arguments typically take the following form:

If p then q; p true, then q true

As mentioned earlier, fuzzy reasoning — also called approximate reasoning — relies on fuzzy rules expressed in natural language through linguistic variables, which we defined previously. A fuzzy rule has the form: If x ∈ A and y ∈ B then z ∈ C, with A, B and C fuzzy sets.

Example: For example: If the quality of the food is delicious, then the tip is high.

The variable “tip” belongs to the fuzzy set “high” to a degree that depends on how valid the premise is — in other words, on the membership degree of the “food quality” variable in the fuzzy set “delicious.” The key idea here is that the more the conditions in the premise are satisfied, the more strongly the suggested output action should be applied.

To determine the truth degree of the fuzzy proposition *the tip will be high*, it is necessary to define the fuzzy implication. As with other fuzzy operators, there is no single standard definition for fuzzy implication; the system designer must choose from a range of existing fuzzy implications or define one manually. Below are two of the most commonly used fuzzy implication definitions:

Table 2: The most commonly Fuzzy implication

Name	Truth value
Mamdani	min (f_a(x), f_bx))
Larsen	f_a (x) × f_b (x)

It is important to note that these two types of fuzzy implication do not generalize classical implication. Other definitions of fuzzy implication do extend the classical form, but they are used less frequently. If we select the Mamdani implication, the fuzzy rule *If the food quality is delicious, then the tip is high* can be evaluated as follows, assuming the food quality is rated at 8.31 out of 10.

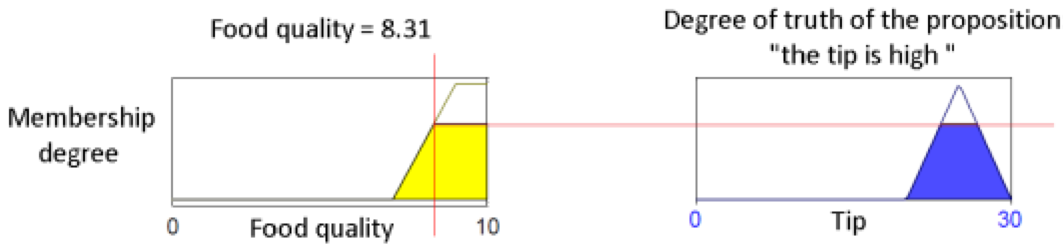


Figure 13 :Example of fuzzy implication

The outcome of applying a fuzzy rule depends on three main factors:

1. The chosen definition of the fuzzy implication.
2. The membership function is defined for the fuzzy set in the rule’s conclusion.
3. The degree of validity of the propositions in the rule’s premise.

Since we have already defined the fuzzy operators AND, OR, and NOT, the premise of a fuzzy rule can also be built from a combination of fuzzy propositions. The complete set of rules in a fuzzy system is called its decision matrix. Below is the decision matrix for our tipping example.

exemple

If the service is bad or the food is awful	then the tip is low
If the service is good	then the tip is average
If the service is excellent or the food is delicious	then the tip is high

Figure 15: illustrates the result for the fuzzy rule If the service is excellent and the food is delicious, then the tip is high, where the service quality is rated at 7.83 out of 10 and the food quality at 7.32 out of 10. This example uses the Mamdani implication and interprets the OR operator with the MAX function.

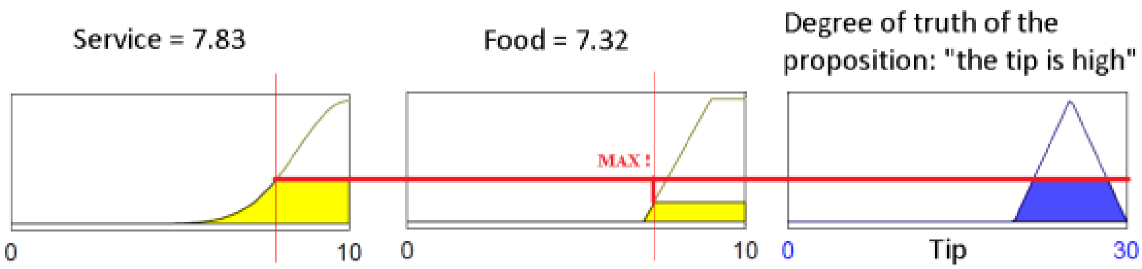


Figure 14: Example of fuzzy implication with conjunction OR translated into a MAX

Next, we will apply all three rules from our decision matrix. As a result, we will get three fuzzy sets representing the tip. These fuzzy sets will be combined using the MAX operator, which is the most commonly used aggregation method. Figure I.15 illustrates this aggregation process.

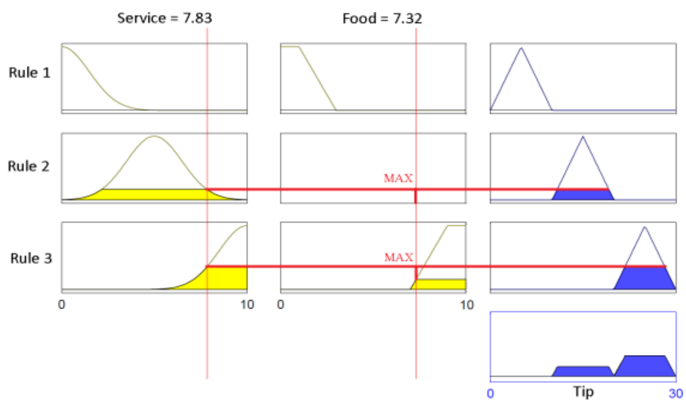


Figure 16: Example of fuzzy implication using the decision matrix

As we can see, it is now necessary to make the final decision — in other words, to determine the actual tip amount, given that the service quality is rated 7.83 out of 10 and the food quality

7.32 out of 10. This final stage, which converts the fuzzy set obtained after aggregating the results into a single concrete decision, is known as defuzzification [9]

I.7.3 Defuzzification:

In order for the fuzzy output set to be applied in practical situations, a defuzzification procedure is necessary. Defuzzification refers to converting a fuzzy set into a crisp set, or turning a fuzzy element into a crisp one. The most widely used defuzzification methods are the Mean of Maximum, the Center of Gravity, and the Height technique [12].

I.7.3.1 The meaning of Maximum (MeOM) Method:

The Mean of Maximum (MeOM) defuzzification method determines the average of the fuzzy output values that reach the highest membership degrees. When dealing with a discrete universe, this defuzzification output z^* can be expressed as: [9]

Where z_i is the control action whose membership functions reach the maximum, and k is the number of such control actions.

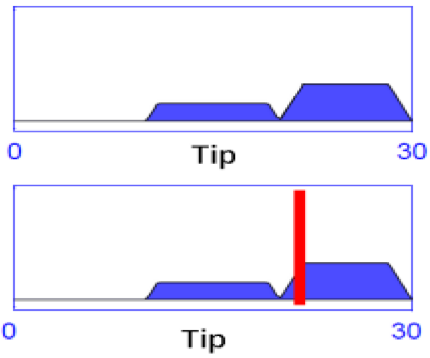


Figure 17: Defuzzification with the method of center of gravity (COG) (last example)

This definition avoids the discontinuities that could appear in the MeOM defuzzification but is more complex and has a greater computational cost.

As illustrated in the two figures that show the application of MeOM and COG defuzzification methods to our example, the choice of defuzzification technique can significantly influence the final decision [9].

I.7.3.3 Height Method (HM):

This method can be divided into two steps. First, the consequent membership function z can be converted into a crisp consequent c_i, where c_i is the center of gravity of μ_z. Then the COG method is applied to the rules with crisp consequences, which can be expressed as: [12].

$$z*=\frac{\sum_{i=1}^{n} z_i \ c_i}{\sum_{i=1}^{n} z_i}$$

I.8 Conclusion:

The main goal of this chapter was to introduce the fundamental concepts of type-1 fuzzy logic, along with the structure of a type-1 fuzzy control system. The basic principles of fuzzy reasoning were explained, showing how new information can be inferred from existing knowledge through linguistic rules.

The concept of fuzzy sets can be applied to many kinds of problems, depending on the type of information, the treatment of imprecision, and the complexity involved. The key advantage of fuzzy systems is their ability to control a system without requiring its precise mathematical model. However, there is no formal procedure for selecting the parameters of a fuzzy controller. The next chapter will provide an overview of type-2 fuzzy logic

CHAPTER II: THE FUZZY LOGIC TYPE 02

II.1. Introduction:

In this chapter, we introduce a new category of fuzzy logic systems known as type-2 fuzzy logic systems, where the membership functions of the antecedents or consequents are type-2 fuzzy sets. The concept of a type-2 fuzzy set was first proposed by Zadeh as an extension of the ordinary fuzzy set (which we referred to in Chapter I as a type-1 fuzzy set). These sets are characterized by membership grades that are themselves type-1 fuzzy sets. Type-2 fuzzy sets are particularly valuable in situations where defining a precise membership function is challenging, making them effective for handling uncertainties.

Very often, the knowledge used to create rules in a fuzzy logic system (FLS) is uncertain. This uncertainty results in rules with uncertain antecedents and/or consequents, which lead to membership functions of the antecedents or consequents also being uncertain.

For example, the words used in linguistic knowledge may have different meanings for different people [13].

A fuzzy relation of a higher type, such as type-2, is considered a way to increase the fuzziness of a relation. According to Hisdal, "increased fuzziness in a description means increased ability to handle inexact information in a logically correct manner." Similarly, John has stated that "type-2 fuzzy sets allow for linguistic grades of membership, thus assisting in knowledge representation, and they also offer improvement on inferencing with type-1 sets."

Type-2 fuzzy sets can represent the uncertainties present in the membership functions of type-1 sets, since these functions depend on available linguistic and numerical information. Linguistic information — for example, expert rules — generally do not specify the exact shape of membership functions. When membership functions are defined or adjusted using numerical data, uncertainties in that data, such as measuring noise, introduce uncertainty into the membership functions. In all these situations, the type-2 framework can incorporate information about both linguistic and numerical uncertainty [14].

I.2. fuzzy logic type 02:

I.2.1. Fuzzy logic type 02 sets:

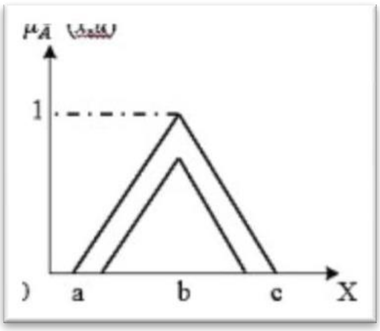
X is a certain universal set. For fuzzy set A, an element membership degree $x \in X$ in this fuzzy set is a real $\mu_A(x)$ from the segment [0, 1].

However, determinng the membership degree of each element x in a fuzzy set A. is not always clear Therefore, the concept was introduced that the membership degree itself could be treated as a fuzzy set, with its support within the interval [0,1] This method has shown to achieve better outcomes in several applied problems [15]

I.2.2. Fuzzy 02 set types:

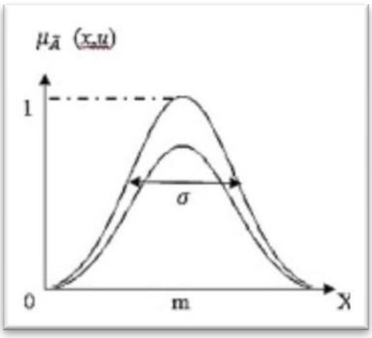
I.2.2.1. Triangular set

The degree of membership of each point is a set triangular type-1 whose definition domain is included in the interval [0, 1].



I.2.2.2. Gaussien set:

In this type of set, the membership degree of each point is represented by a type-1 Gaussian set, whose domain lies within the interval [0,1][0, 1][0,1]. It is important to note that the primary membership function does not necessarily have to be Gaussian as well.



I.2.2.4. Interval set:

In this type of sets, the degree of membership of each point is an ordinary set whose domain of definition is included in the interval [0 1]. In this case, all secondary memberships are equal to 1. Noting that although each degree of a type-2 interval set is an ordinary set, the set itself is type2, because the membership degrees are set and not ordinary numbers.

I.2.3. Representation of type-2 fuzzy sets:

A type-2 membership function can be seen as a function of two variables. For each x and the primary membership degree μ1, we have a membership secondary, which is an ordinary number noted μ2 (Fig II.1)

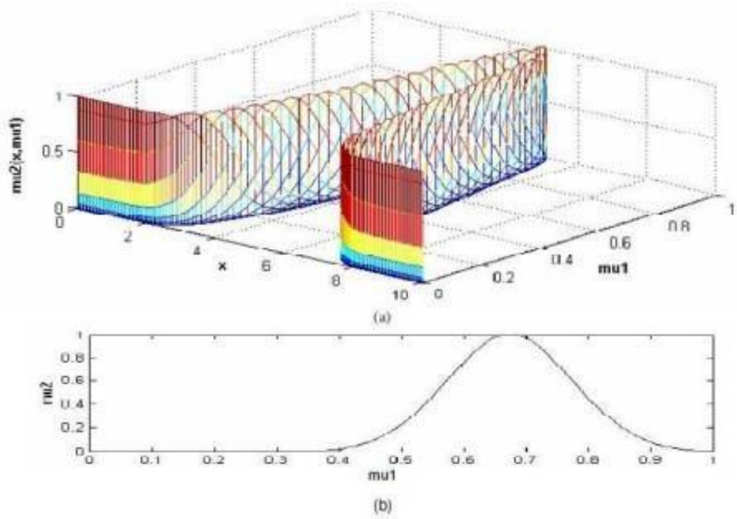


Figure 18: a) Three-dimensional representation of a Gaussian type-II fuzzy set
b) Corresponding fuzzy membership degree for x = 4.25

Thus, a type-2 membership function can be represented as follows:

$$\mu_2(x, \mu_1): X \times [0, 1] \rightarrow [0, 1] \quad \text{(II.13)}$$

Where X is the space of entries x. The figure (Fig.II.1 a) is a three-dimensional representation of a type-2 Gaussian fuzzy set, having a principal membership function Gaussian, and the figure (Fig.II.1 b) represents the Gaussian-type fuzzy membership degree corresponding to x =4.25.

II.2.4. Type 2 fuzzy set operations:

We have seen that the degree of membership of a type-2 fuzzy set is a type-1 fuzzy set by therefore, to perform operations such as union and intersection on sets fuzzy type-2 we need to be able to perform the **t-conorm** and the **t-norm** between two sets type-1. This is done using the extension principle of Zadeh. [16]

Table 3: Main t-norms and T-conorms

	t-normes	t-conorm
Zadeh (1972)	min(x, y)	max(x, y)
Bendler Kahout (1980)	$x.y$	$x+y-x.y$
Lukasiewicz, Giles (1976)	max($x+y$-1,0)	max($x+y$, 1)
Weber (1983)	$\begin{cases} x \; si \; y = 1 \\ y \; si \; x = 1 \\ 0 \; otherwise \end{cases}$	$\begin{cases} x \; si \; y = 1 \\ y \; si \; x = 1 \; otherwe \end{cases}$
Hamacher (1978) **$\gamma > 0$**	$\dfrac{x, y}{\gamma + (1-\gamma)(x + y - x.y)}$	$\dfrac{x + y - (2-\gamma)x, y}{1-(1-\gamma)x.y}$
Dubois and Parade (1986) $\alpha \in$ [0,1]	$\dfrac{\mathrm{x, y}}{\max(\mathrm{x, y}, a)}$	$\dfrac{\mathrm{x + y + x, y} - \min(\mathrm{x, y}, 1 - a)}{1 - (1 - \mathrm{y})\,\mathrm{x.y}}$

II.3. Type-2 fuzzy logic systems:

The structure of a T2FLS, as presented in Fig.II.3, is quite similar to a T1FLS. The only difference is that the antecedent and/or consequent sets in a T2FLS are type2, so that each rule output set is type-2. There are five principal parts in a T2FLS: Fuzzifier, rule base, inference engine, type-reducer and defuzzifier. [17].

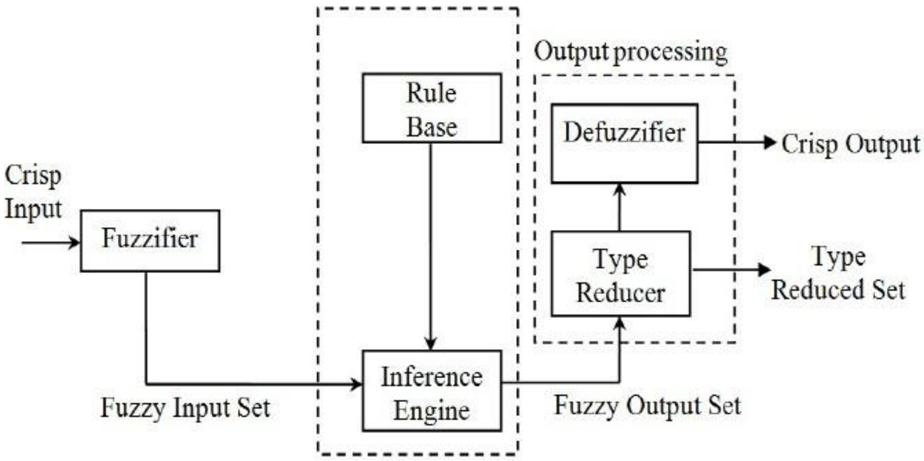


Figure 19: Type-2 fuzzy logic system diagram

The type-reducer carries out a type-reduction process, which represents an expanded form of

T1 defuzzification. Through type reduction, a T1 set is produced from the output set of the T2 rules. This generated T1 set is referred to as the type-reduced set. Afterwards, the type-reduced set can be defuzzified to deliver a precise output. In a T2FLS, the type-reduced set reflects the possible range of variation in the crisp output caused by uncertainties in either the antecedents or the consequences. [17]

II.3.1. Fuzzifier:

The fuzzifier maps a crisp input vector with n fuzzy input sets, which can be type-2 (T2) fuzzy input, sets $\bar{A}_x$ en general. However, we will consider the singleton fuzzification as it is fast to compute, and thus suitable for real time applications. In the singleton fuzzification, the input fuzzy set has only a single point of nonzero membership Thus, $\bar{A}_x$ is a T2 fuzzy singleton if $\bar{\mu} = 1$ for $x_i = x_i^0$ and $\mu_{\tilde{A}x}$ (xi) = 0 for all $x_i \neq x_i^0$, (i = 1, 2 ...n). [17]

II.3.2. Rule base:

For an interval type-2 fuzzy logic system (IT2FLS), the j-th rule can be written as:

$$R^{j:} \text{ If } x_1 \text{ is } V_1^j \text{ and } \dots \text{ and } x_n \text{ is } V_n^j$$

$$\text{Then } y \text{ is } \Theta_j,\ j = 1 \dots r \qquad \text{(II.22)}$$

Where V^j are antecedent type-2 sets, y ∈ Y is the output, Θ_j are consequent T2 sets. The rule in (II.22) represents a T2 fuzzy relation between the input and the output spaces on the FLS.

II.3.3. Fuzzy inference engine:

In the considered IT2FLS, we will use the product operation. In an IT2FLS with minimum or product tnorm, the firing interval Vj of the j-th rule is an interval type-2 set, which is determined by its left most point and right most point $\underline{v}^j$ and v^j as:

$$V^j(X^0) = [\underline{v}^j (X^0), \bar{v}^j (X^0)] \equiv [\underline{v}^j, \bar{v}^j]$$

Where X0 is the instantaneous value of X. Accordingly, the firing interval bounds for the j-th rule of an IT2FLS with n inputs $\underline{v}j$ and v^j, can be written as:

$$\underline{v}j = \underline{\mu}_{v1}^j(x_1^0) \dots \underline{\mu}_{vn}^j(x_n^0) = \prod_{i=1}^n \underline{\mu}_{vi}^j(x_i^0)$$

$$\bar{v}^j = \bar{\mu}_{v1}^j(x_1^0) \dots \bar{\mu}_{vn}^j(x_n^0) = \prod_{i=1}^n \bar{\mu}_{vi}^j(x_i^0)$$

II.3.4. Type reduction:

This process is known as type-reduction because it transforms the type-2 output sets from the inference engine into a type-1 set. The resulting type-reduced sets are then defuzzified to produce a crisp output. This approach takes less computation time compared to the centroid method and helps to avoid the difficulties that may arise with other techniques. Type reduction using the COS method is given by:

Type reduction using the COS method is given by:

$$Y_{cos} = \int_{\theta^1} \cdots \int_{\theta^r} \cdots \int_{v^1} \cdots \int_{v^r} \frac{\sum_{j=1}^{r} v^l}{\sum_{j=1}^{r} v^l \theta^j} d\theta^1 \cdots d\theta^r dv^1 \cdots dv^r$$

Where YCOS is the interval set determined by two end points y*l*(X) and yr(X),

$\theta^j \in \Theta^j = [\theta_l^j, \theta_r^j]$ is the type-2 interval consequent set, and $v^j \in V^j = [\underline{v}^j, \overline{v}^j]$ is the firing interval.[16]

II.4. Advantages and inconveniences of Fuzzy Logic System:

II.4.1 Advantages of Fuzzy Logic System:

- The structure of fuzzy logic systems is simple and easy to grasp.
- Fuzzy logic is commonly applied for both commercial and practical uses.
- Within artificial intelligence, fuzzy logic supports the control of machines and consumer devices.
- Although it might not deliver perfectly accurate reasoning, it offers the only reasoning that is acceptable in some situations.
- It assists in managing uncertainty in engineering applications.
- It is generally robust since it does not depend on precise input values.
- It can be programmed to respond even if a feedback sensor fails.
- The system can be modified easily to enhance or change its performance.
- Low-cost sensors can be used, which helps to reduce the total cost and complexity of the system.
- It presents a highly effective way to address complex problems. [18]

II.4.2 inconvenience of Fuzzy Logic Systems:

- Fuzzy logic does not always produce highly accurate outcomes, so the results rely on assumptions, which may limit their acceptance.
- Fuzzy systems lack the abilities of machine learning and pattern recognition found in neural networks.
- Validating and verifying a fuzzy knowledge-based system requires extensive testing with physical hardware.
- Defining precise fuzzy rules and membership functions can be a challenging process.

- Sometimes fuzzy temporal logic is mistaken for probability theory and its related terminology. [18]

II.5. Fuzzy Logic in Artificial Intelligence:

Fuzzy logic is applied in natural language processing as well as in several demanding artificial intelligence applications. Fuzzy logic systems produce logical outputs in response to uncertain, noisy, distorted, and incomplete fuzzy inputs. In artificial intelligence, fuzzy logic represents a reasoning method that resembles human thinking. This approach imitates human decision-making and should consider all aspects of the problem, delivering a digital output in the form of YES or NO.

The fundamental logic system uses an n-valued logic structure, applying a scale to the input states and producing outputs based on both the state of the input and the extent of its changes. (While these outputs may technically be TRUE or FALSE, they are interpreted by humans as YES, NO, or partially YES/NO). [19] For this reason, the inventor of fuzzy logic systems has included every possible input logic to allow the system to generate a clear output. [19] As a result, fuzzy logic is highly suitable for the following applications: [20]

- Supporting engineering decisions in cases with unclear certainties or uncertainties, or with vague data, as is common in natural language processing technologies
- Regulating and controlling machine behavior based on multiple input variables, such as in temperature control systems

II.6 . Conclusion:

This chapter has been dedicated to the introduction of the type-2 fuzzy logic, where we have presented the theoretical foundation and the basic notions of this logic. This logic is very effective in circumstances where it is very difficult to determine the exact membership functions for a fuzzy system; therefore, this new logic allows us to incorporate uncertainties in the rules, which will act positively on the output of the considered system.

CHAPTER III: WOOD AND BERRY DISTILATION COLUMN MODEL

CHAPTER III: WOOD AND BERRY DISTILATION COLUMN MODEL

III .1. Introduction to Distillation

Distillation is among the most commonly used separation techniques in chemical engineering, mainly applied to divide liquid mixtures by taking advantage of differences in their boiling points. This operation is essential in sectors like petrochemicals, pharmaceuticals, and food processing. During distillation, a liquid mixture is heated until it reaches its boiling point, producing vapor that rises through a distillation column. The vapor is then condensed into a liquid, which becomes enriched with the more volatile substance of the mixture. This process is carried out over several stages within the column, allowing for a greater separation of the components. The effectiveness of the separation depends on various factors, such as the relative volatility of the substances and the number of stages inside the column.

In a typical distillation column, two key products are produced: the distillate and the bottoms. The distillate, often called the light key, is the more volatile component, while the bottoms, known as the heavy key, are the less volatile component. The separation process is controlled by the vapor-liquid equilibrium, which varies with changes in temperature and pressure. The degree of separation relies on the reflux ratio (the proportion of condensed liquid sent back to the column compared to the liquid collected as distillate) and the heat supplied by the reboiler located at the base of the column.

Distillation is crucial in converting crude oil into products like gasoline, diesel, and other valuable fuels. As noted by Seader et al. (2011), distillation columns consume a significant share of the total energy used in industrial operations, and improving their control can lead to major energy savings. Ongoing improvements in distillation technology and control systems aim to enhance separation efficiency while lowering energy demands in these processes. [21]

CHAPTER III: WOOD AND BERRY DISTILATION COLUMN MODEL

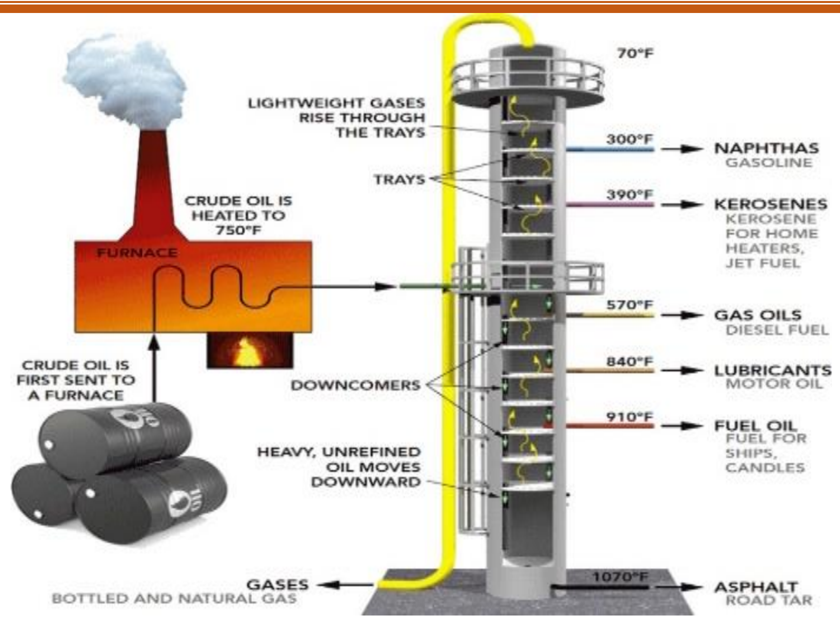


Figure 20: distillation column

III .2. Introduction to Distillation Columns:

Distillation columns represent some of the most vital and commonly used unit operations within the chemical and petrochemical industries. Their primary role is to separate liquid mixtures into their individual components by exploiting differences in volatility. This separation process depends on a sequence of repeated vapor–liquid equilibrium stages, which gradually enrich the vapor phase with the more volatile components while concentrating the less volatile components in the liquid phase, achieving the targeted product purities. According to Luyben, distillation makes up nearly 40% of the total energy use in many petrochemical plants, highlighting its critical economic and environmental impact. [22]

In a standard distillation column, the feed mixture enters around a middle tray, where it interacts with rising vapor and falling liquid across a series of trays or packing elements. The vapor that exits from the top of the column is condensed and partly sent back as reflux, while the liquid product is taken from the bottom after partial vaporization in a reboiler. Maintaining energy and mass balances along the entire column creates a complex dynamic system, which is highly sensitive to changes in the feed's properties, composition, and flow rates. [23]

III .3. Types and Classification of Distillation Columns:

Distillation columns can be categorized based on several factors:

- **Number of feeds and products:** for example, binary columns with a single feed and two products, compared to multicomponent columns.
- **Mode of operation:** continuous versus batch distillation processes.
- **Internal design:** tray columns (including sieve trays and valve trays) versus packed columns (using either structured or random packing).
- **Level of separation difficulty:** such as simple distillation, azeotropic distillation, or extractive distillation.

In petrochemical facilities, binary distillation is still the most widely applied method, as it provides a foundation for separating two main products. Nevertheless, even binary columns face complex multivariable interactions and significant disturbances. The effectiveness of these columns relies on precisely managing the reflux ratio, reboiler heat input, and internal flow distributions, which calls for advanced control methods beyond traditional single-loop PID systems. [24].

III .4. Binary Distillation Columns: Characteristics and Challenges

A binary distillation column is designed to separate two components, typically labeled as the light key (the more volatile compound) and the heavy key (the less volatile compound). The main goals usually include obtaining high purity for both the distillate (top product) and the bottoms (bottom product), while also reducing energy usage. Nonetheless, binary columns exhibit strong interactions: modifying the reflux ratio not only impacts the composition of the distillate but can also alter the composition of the bottoms due to internal mass and energy balance relationships. [25]

Other difficulties faced with binary distillation processes include:

- Nonlinear behavior resulting from phase equilibrium dynamics
- Process disturbances such as changes in feed composition or flowrate fluctuations
- Time delays and dead times in measurements of temperature or composition.
- Challenges in optimizing energy usage while meeting purity targets.

Given these obstacles, binary columns have for many years been used as a benchmark for testing advanced control strategies, including multivariable control, adaptive control, and fuzzy logic controllers. [26]

III .5. Wood and Berry Model: Background and Importance

In 1973, Wood and Berry introduced a simplified yet influential dynamic model of a binary distillation column, intended to support research in control system design. [27] The Wood and Berry model represents a two-input, two-output (TITO) system, describing how the reflux flow rate and the heat duty of the reboiler influence the compositions of the distillate and the bottoms. Its structure captures the key dynamic characteristics of a binary column, such as:

- strong loop interactions (cross-coupling)
- relatively large time constants and delays
- nonlinear behaviors associated with phase-change operations

These features have established the Wood and Berry model as a widely accepted benchmark for assessing multivariable control techniques. It is frequently employed to evaluate decentralized control, model predictive control (MPC), and advanced intelligent controllers like fuzzy logic. [28]

Mathematically, the Wood and Berry distillation column is typically expressed in transfer function matrix form, where:

- Represents the distillate composition
- Represents the bottoms composition
- Refers to the reflux flow rate
- Denotes the reboiler heat duty

Time delays, expressed through exponential terms, model the physical transport delays occurring within the column. [29].

III .6. Applications and Control Significance of the Wood and Berry Model

Even after fifty years, the Wood and Berry model continues to hold importance because it captures the genuine challenges of controlling binary distillation processes, such as:

- Strong interactions between process variables
- The existence of dead times
- High sensitivity to disturbances

As a standard benchmark, this model has been widely used to test fuzzy logic controllers, decentralized control strategies [30], and model predictive control (MPC) approaches [31]. Additionally, thanks to its moderate order, it is practical to simulate using MATLAB or other control software, which has made it a popular choice in both academic research and educational settings.

Numerous researchers have shown that conventional PID controllers have difficulty achieving tight composition control on the Wood and Berry system because of its multivariable interactions. This challenge has encouraged the adoption of advanced and intelligent control methods, including Type-1 and Type-2 fuzzy controllers, which can handle uncertainty and cross-coupling more effectively. [32]

Process Model:

The well-known Wood and Berry distillation column represents a classic MIMO (multiple-input multiple-output) system, characterized by strong variable interactions and considerable time delays. The plant model is divided into a 2-input, 2-output section and a 1-input, 2-output section. The first section describes the transfer function matrix that links the manipulated variables to the system outputs, while the second section describes the transfer function matrix connecting disturbances to the outputs. A schematic diagram of the modeled column is presented in Figure below

CHAPTER III: WOOD AND BERRY DISTILATION COLUMN MODEL

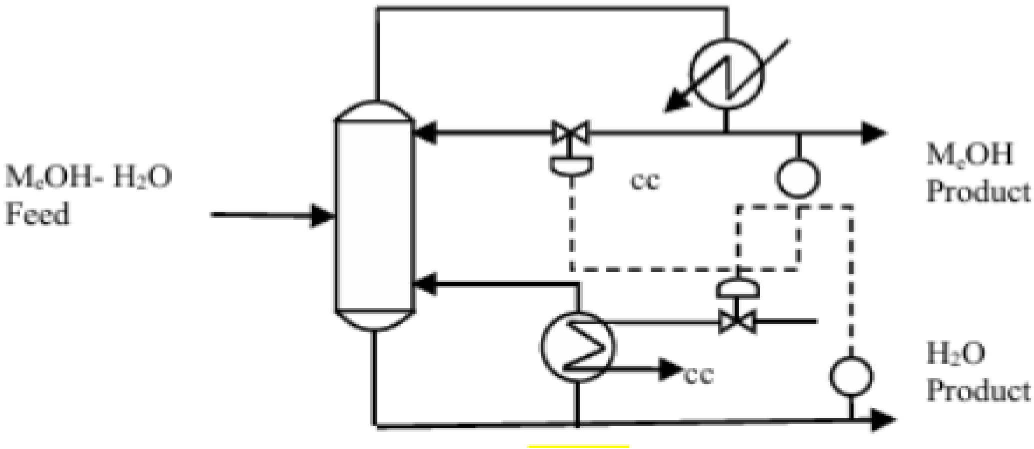


Figure 21: wood/berry distilation column

The transfer function form of the model reported by wood/berry is :

$$\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} \frac{12.8e^{-s}}{16.7s+1} & \frac{-18.9e^{-3s}}{21.0s+1} \\ \frac{6.6e^{-7s}}{10.9s+1} & \frac{-19.4e^{-3s}}{14.4s+1} \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} + \begin{bmatrix} \frac{3.8e^{-8s}}{14.9s+1} \\ \frac{4.9e^{-3s}}{13.2s+1} \end{bmatrix} d(s)$$

Where $y_1(s)$ is the mole fraction of methanol in the tops; $y_2(s)$ is the mole fraction of methanol in the bottoms; $u_1(s)$ is the reflux flow rate; $u_2(s)$ is the steam flow rate and $d(s)$ is the feed flow rate. The time constant and delays are units of minutes. For controller design purpose [33]

Summary and Transition to Advanced Control Strategies

In summary, distillation columns are vital in the petrochemical industry but pose serious challenges due to their nonlinear, multivariable, and highly interactive dynamics. Binary distillation systems, in particular, have served as a classical benchmark for developing and testing modern control approaches. The Wood and Berry model, with its realistic yet mathematically tractable structure, has played a critical role in this domain. Its ability to capture essential dynamics, including time delays and cross-coupling, makes it ideal for evaluating new controllers before their industrial implementation.

This chapter sets the stage for the next discussion, where advanced control methods — including fuzzy logic control, both Type-1 and Type-2, and PID controllers — will be explored in the context of the Wood and Berry distillation column.

CHAPTER IV: SIMULATION AND RESULTS

IV.1. Introduction

This chapter is devoted to the representation of the results obtained while using the PID regulator and fuzzy logic (type 1 and 2) for the control of systems mentioned in the previous chapter.

IV.2. column distillation wood and berry

This is the mimo system we will be controlling:

IV.2.1. control using PID regulator:

Here we have brought two PID controllers and connected them to the system monitor. System modeling using Simulink in MATLAB. and we obtain figure IV.1.

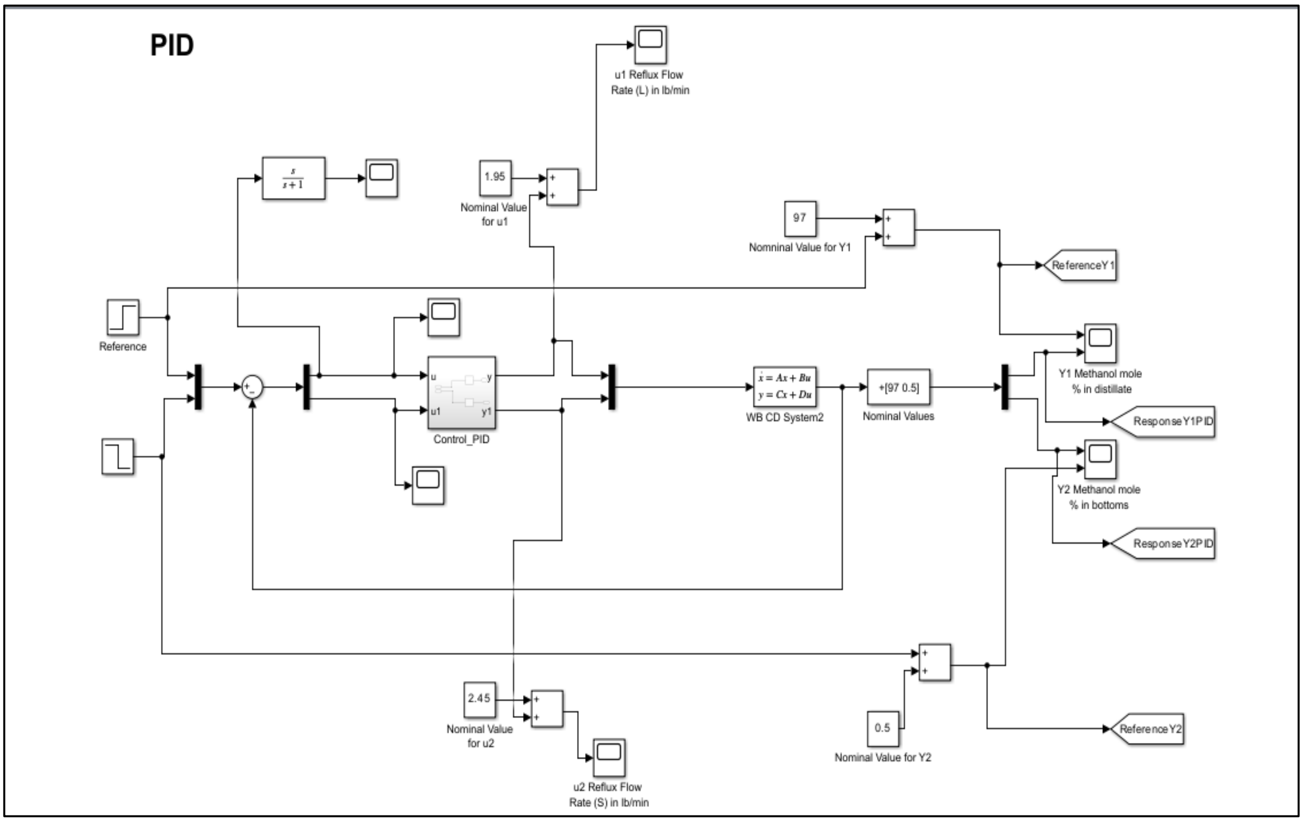


Figure 22: Block diagram of column wood and berry using control PID

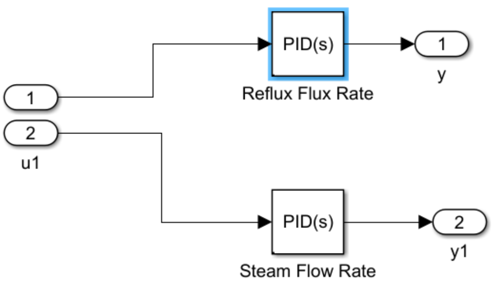


Figure 23: Sub-system of block diagram the regulator PID

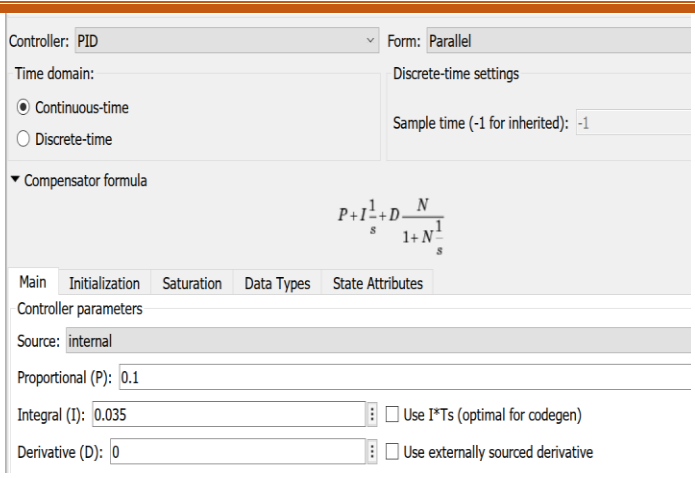


Figure 24: setting the pid_1

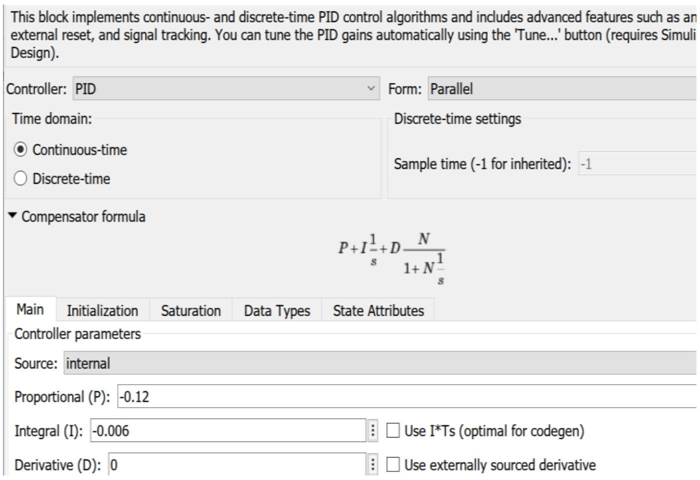


Figure 25: setting the pid_2

- ## Simulation result

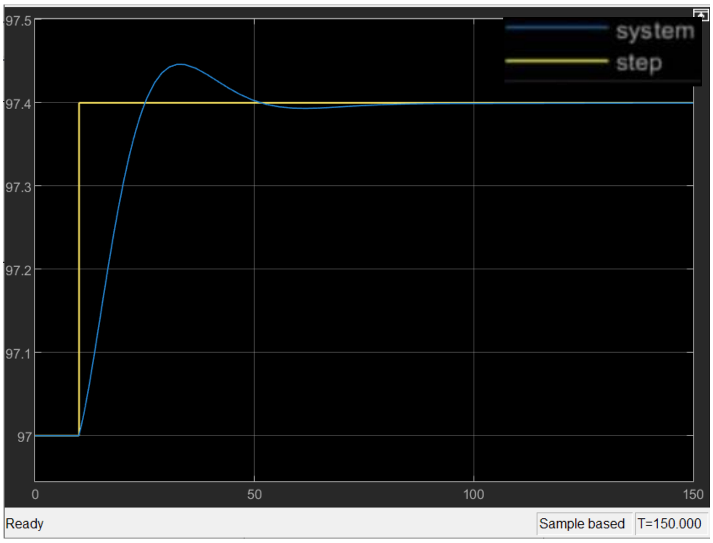


Figure 26: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

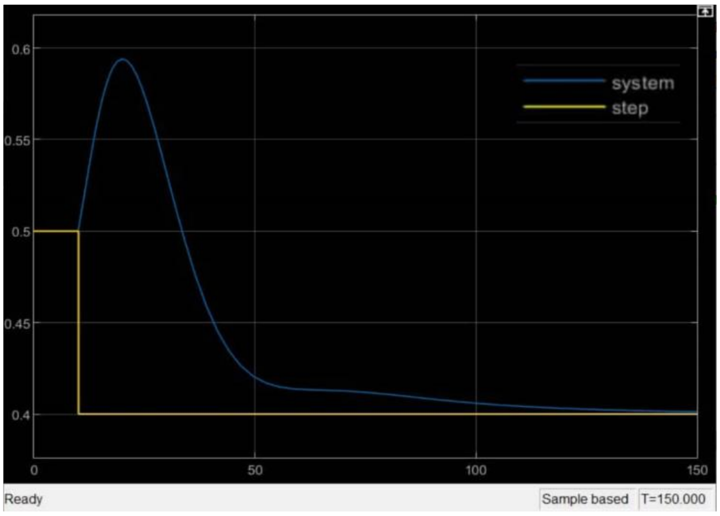


Figure 27: Variations of the Y2 Methanol mole % in bottom and the reference as a function of time

✓ Interpretation:

From Figure 26 and Figure 27, it can be seen that the output does not follow the setpoint (slow and inaccurate response), which means that the setpoint tracking property is not achieved. There are overflows.

In case of perturbation

- **Simulink result**

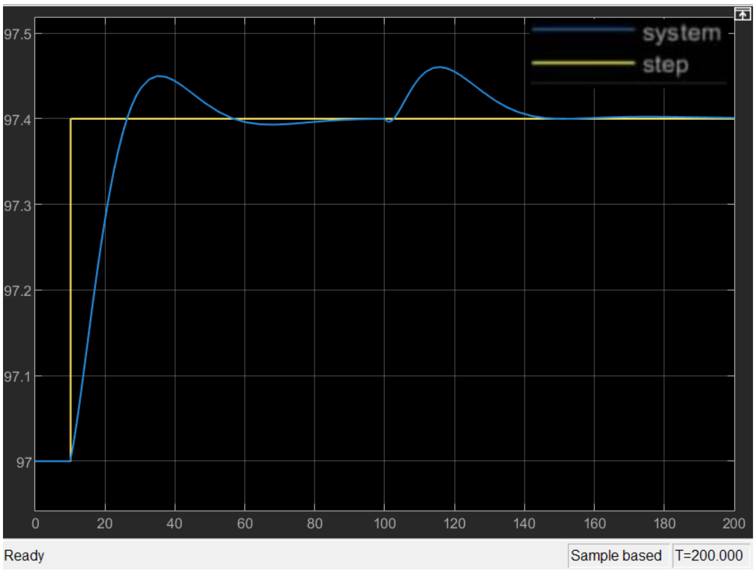


Figure 28: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

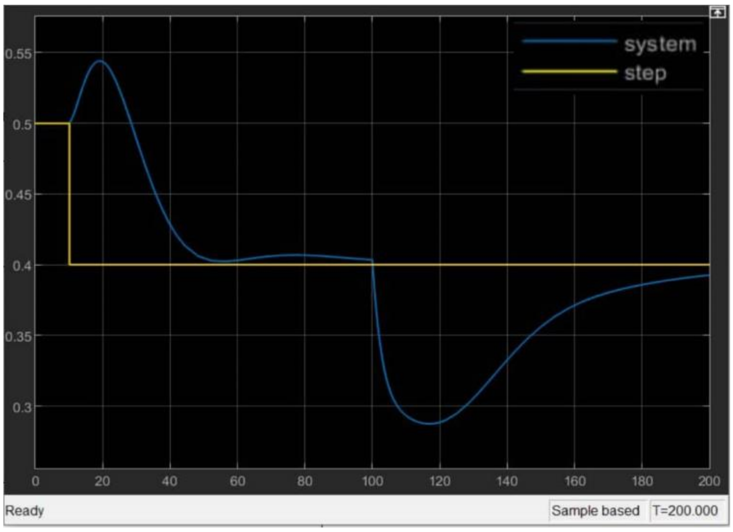


Figure 29: Variations of the Y2 Methanol mole % in bottom and the reference as a function of time

- ✓ **Interpretation:**

From figure 28 and 29 it can be seen that the outputs did not respond fast when exposed to perturbation it took a long time to track the set point.

IV.2.2. Application of fuzzy logic control type-1:

Here we have added fuzzy logic type 1 to the system. System modeling using Simulink in MATLAB. and we obtained figure 30.

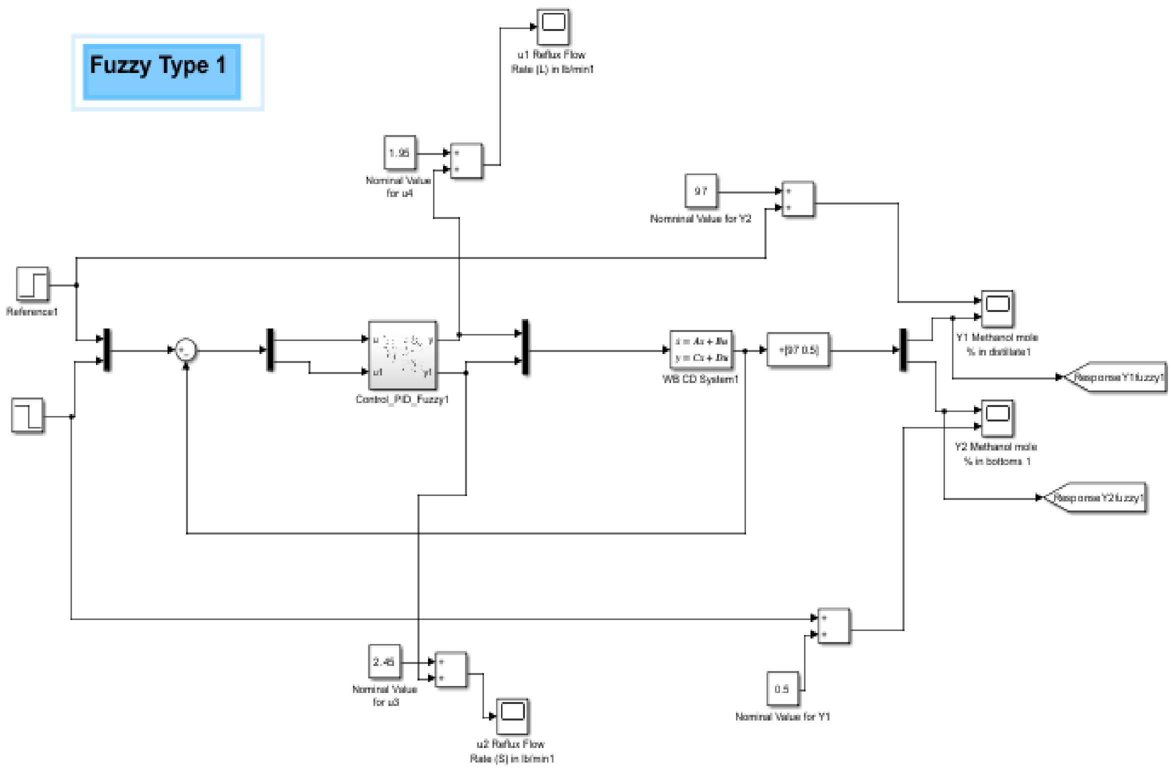


Figure 30:Block diagram of Fuzzy logic type 1 applied to column wood and berry

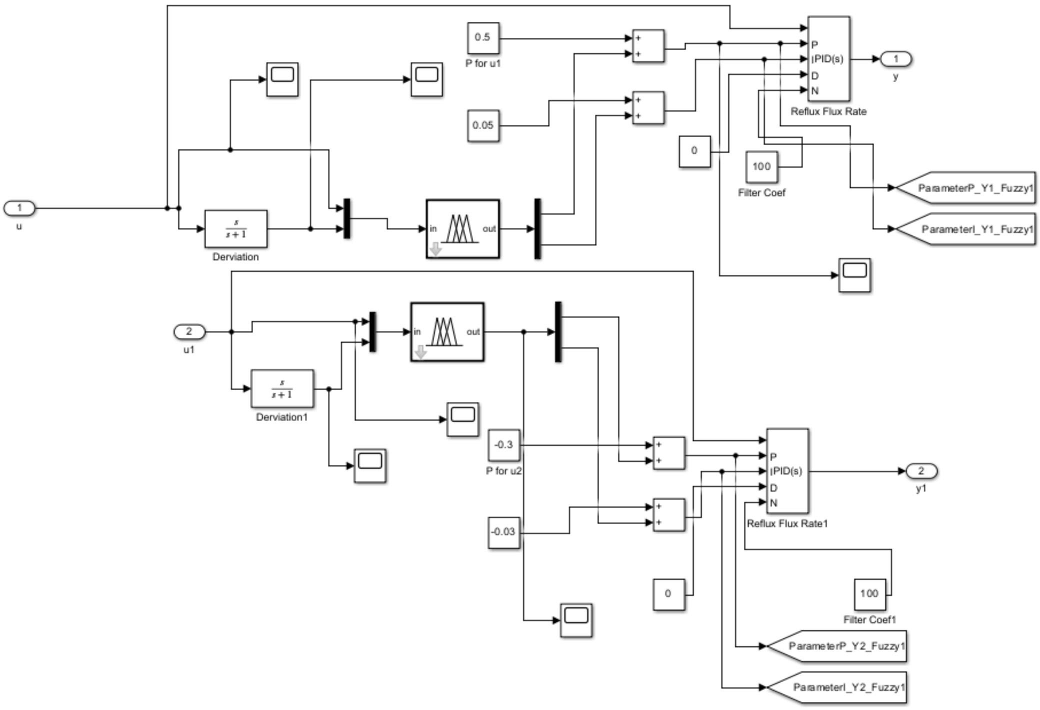


Figure 31: Sub-system of block diagram the Fuzzy PID

Fuzzy Logic Toolbox Graphical Interface:

To control this system, we will design a fuzzy MIMO controller. The "fuzzy" command opens the FIS editor graphical interface, where the entire fuzzy system can be defined. By default, the interface proposes inputs and outputs using the Mamdani method. The (AND) and (OR) operations are performed using (min) and (max), respectively, inference is performed using min, rule clustering is performed using max, and defuzzification is performed using the center of gravity (center) method.

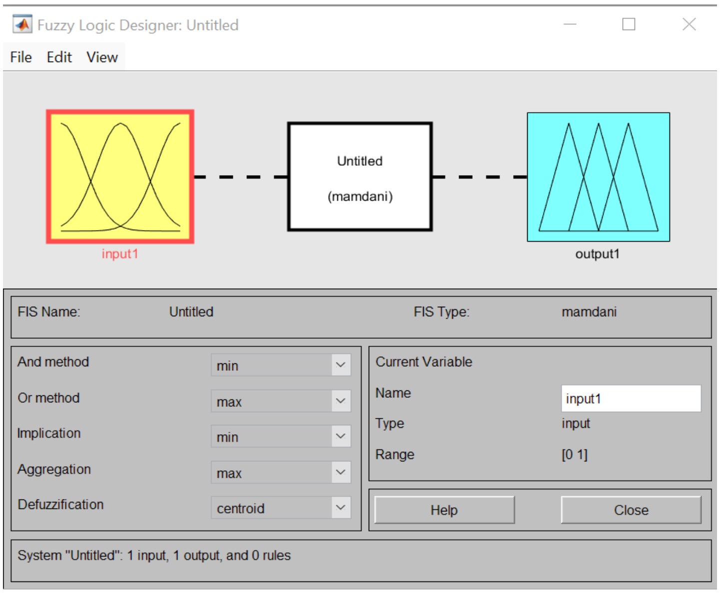


Figure 32: Graphical interface FIS Editor

Fuzzy1_PID_y1:

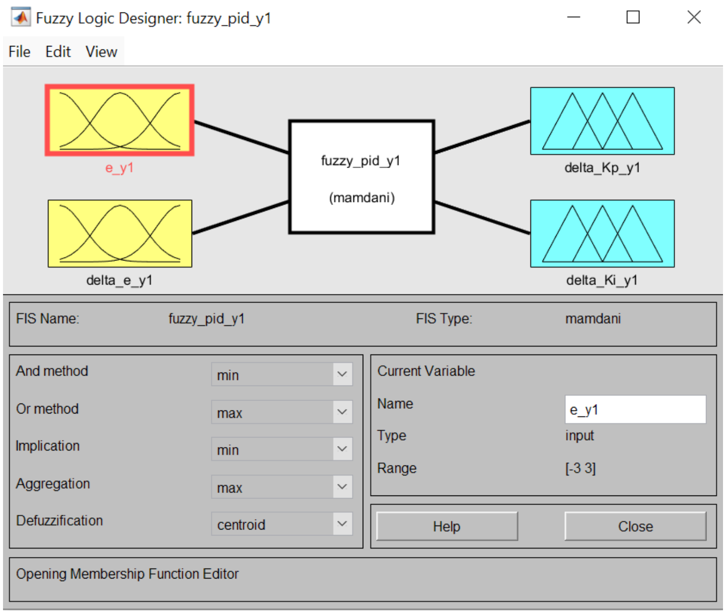


Figure 33: Graphical interface FIS Editor Fuzzy1 y1

The membership function universes for error (e) were normalized by [-3 3] and for control (flow) was [-3 3]. Figure 34, Figure 35, Figure 36, Figure 37, show the input and output membership functions for the fuzzy controller used to solve this problem, respectively.

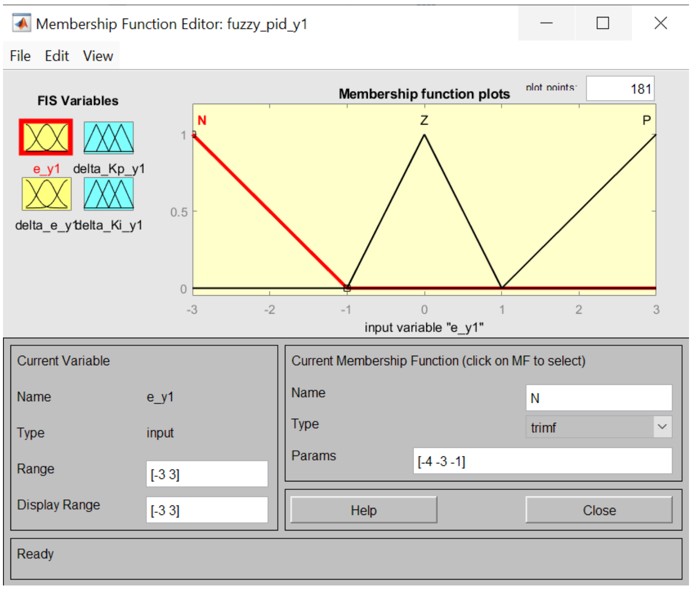


Figure 34: 1st input MFS

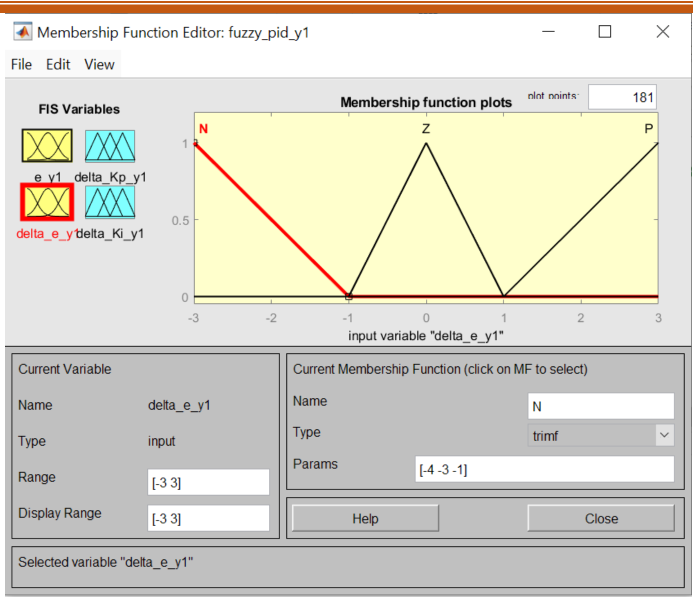


Figure 35: 2nd input MFS

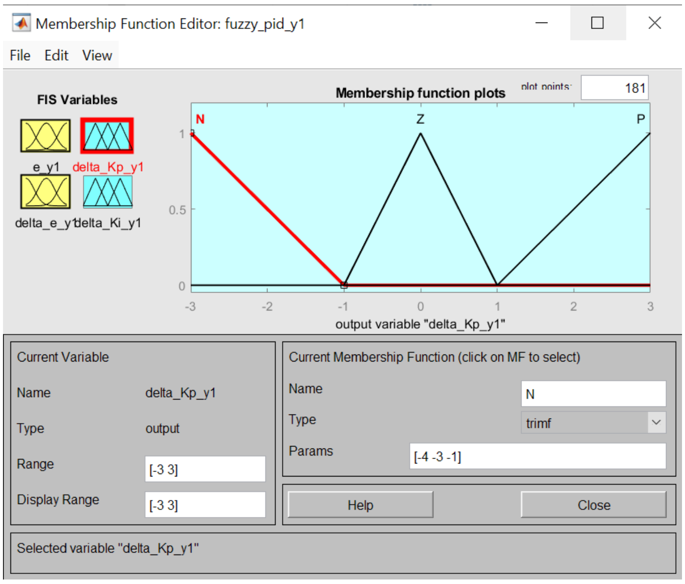


Figure 36: 1st Output MFS

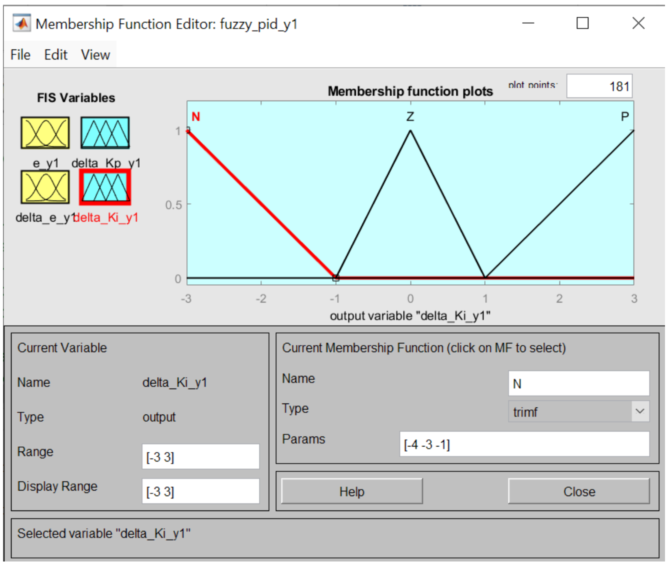


Figure 37: 2nd Output MFS

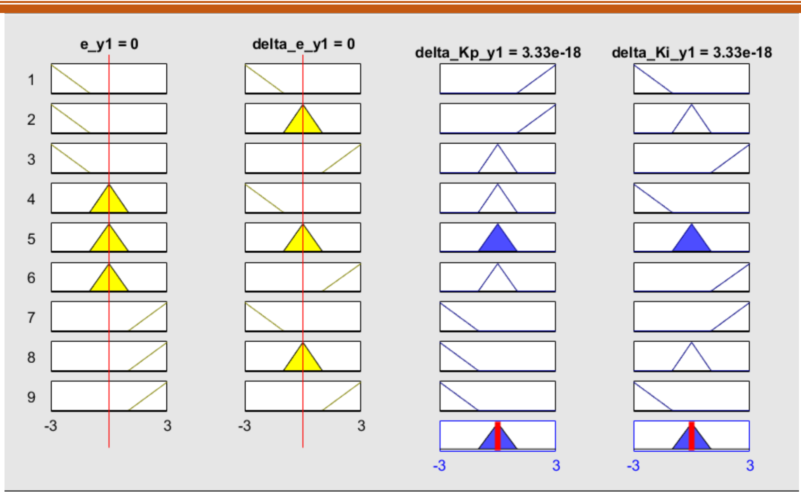


Figure 38: Rules view

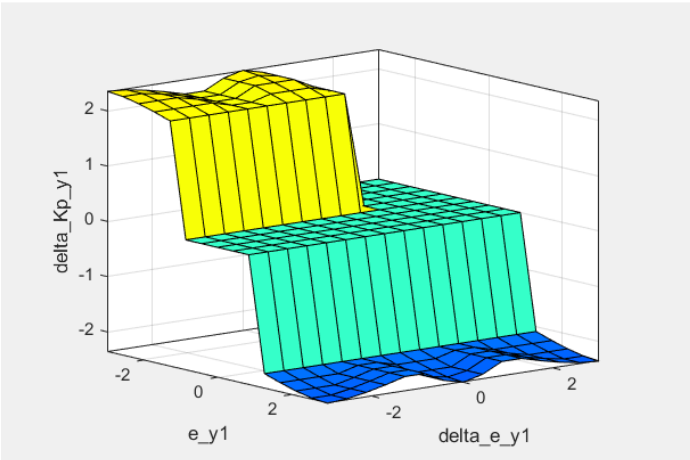


Figure 39: Surface view

Fuzzy1_PID_y2:

The membership function universes for error (e) were normalized by [-3 3] and for control (flow) was [-3 3]. Figure IV.19, Figure IV.20, Figure IV.21, Figure IV.22, show the input and output membership functions for the fuzzy controller used to solve this problem, respectively.

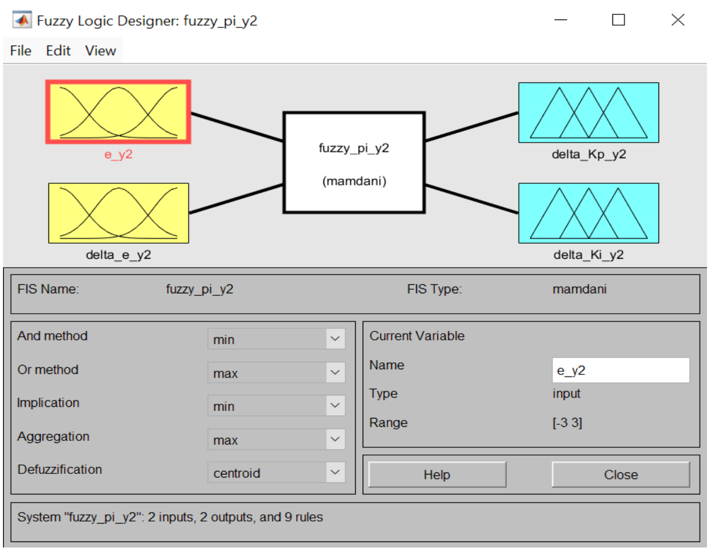


Figure 40: Graphical interface FIS Editor Fuzzy1 y2

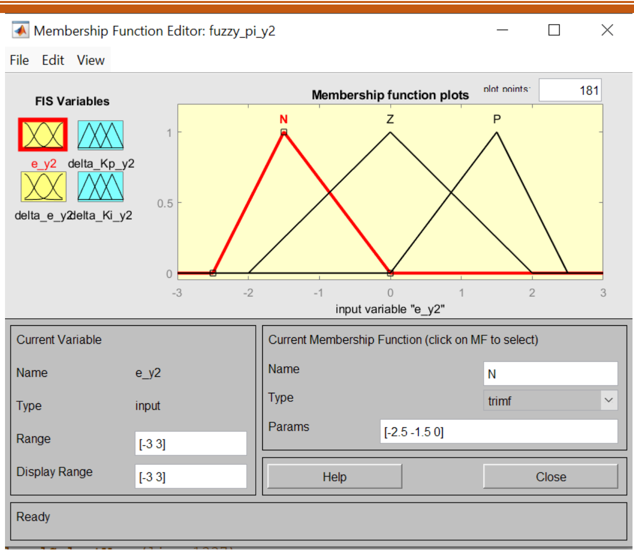


Figure 41: 1st input MFS

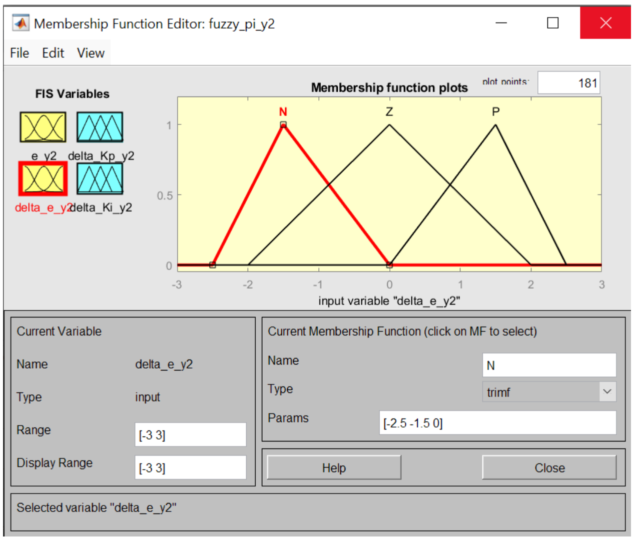


Figure 42: 2nd input MFS

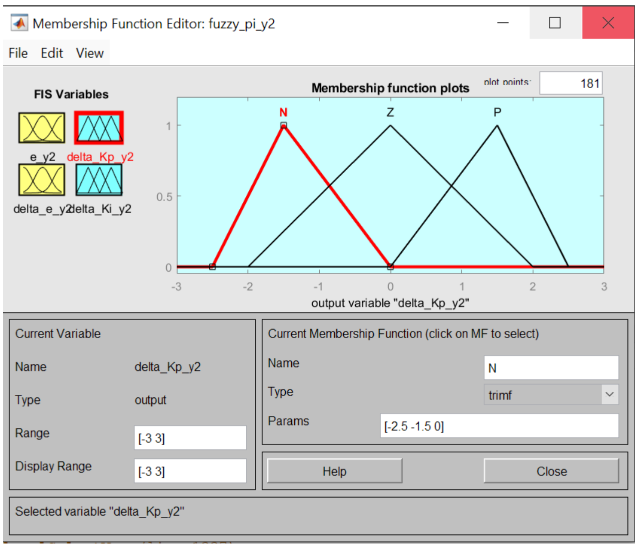


Figure 43: 1st Output MFS

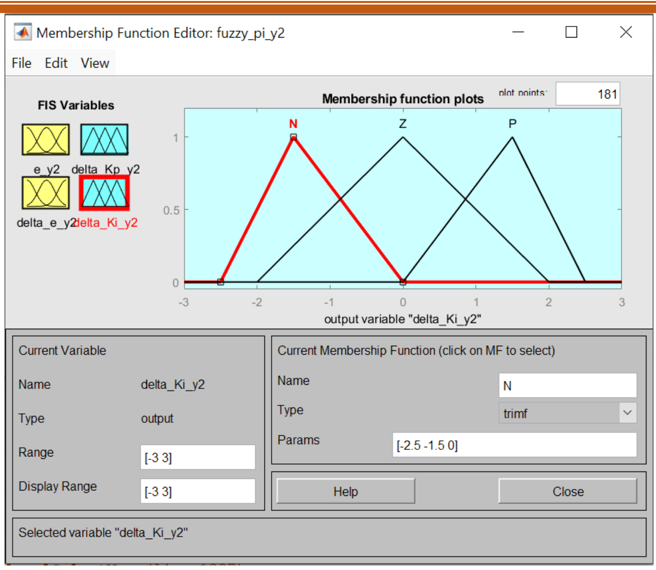


Figure 44: 2nd Output MFS

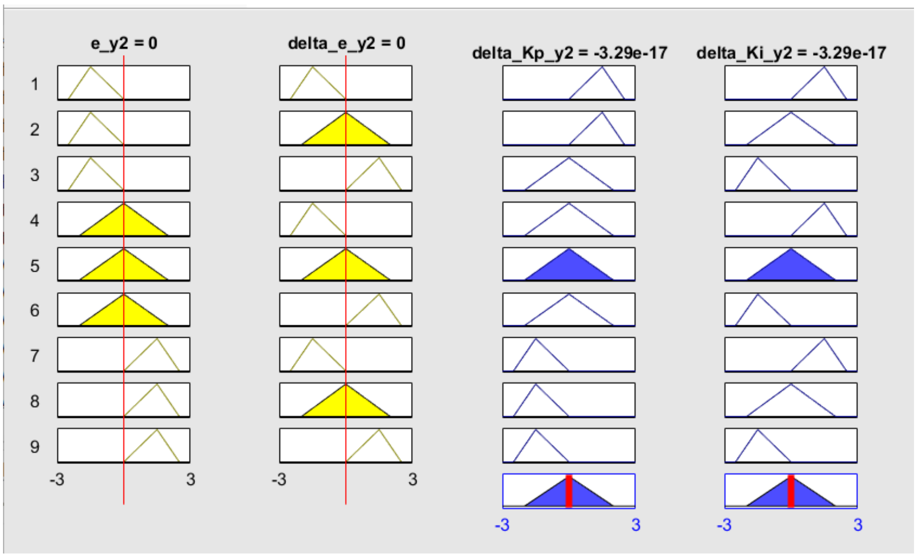


Figure 45: Rules view

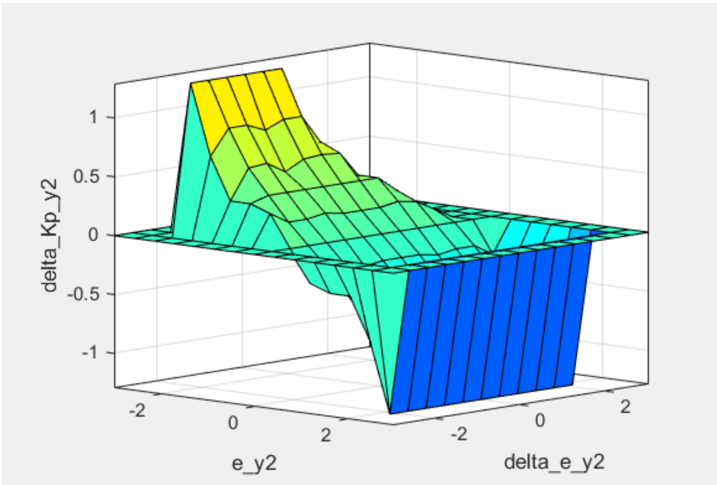


Figure 46: Surface view

- **Simulation result**

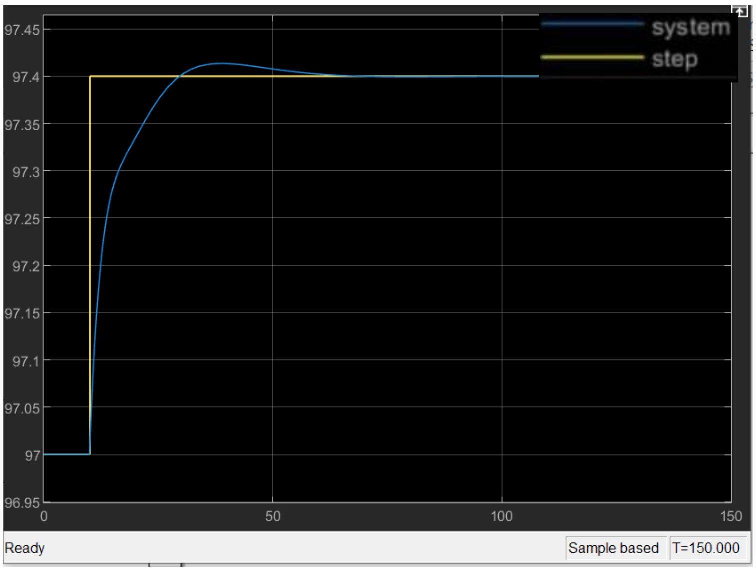


Figure 47: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

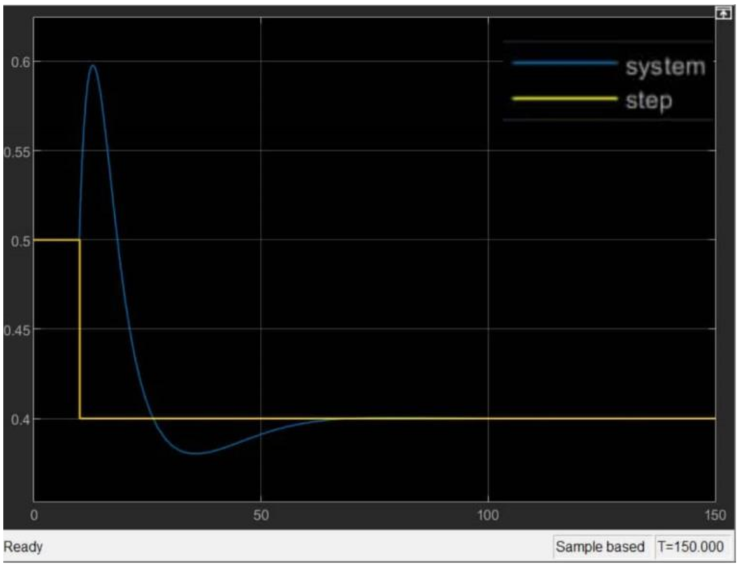


Figure 48: Variations of the Y2 Methanol mole % in Bottom and the reference as a function of time

✓ **Interpretation:**

From Figure 47, and 48 it can be seen that the system has a faster response and better precision when applying Fuzzy Logic type 1, which means that the setpoint tracking feature is achieved, with few excesses.

In case of perturbation:

- **Simulink result**

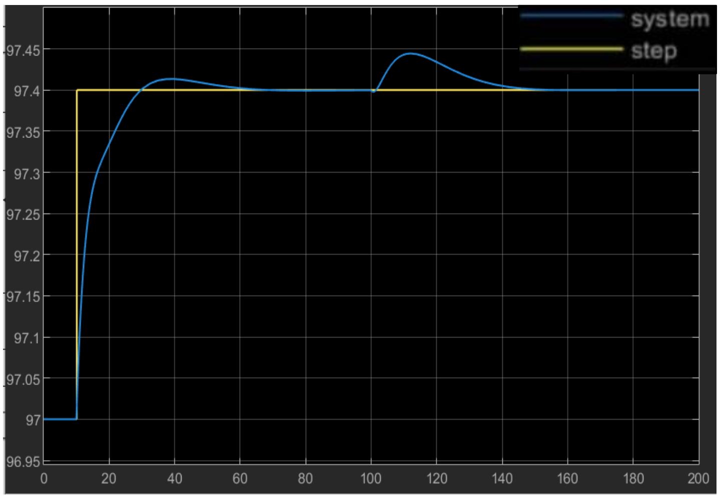


Figure 49: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

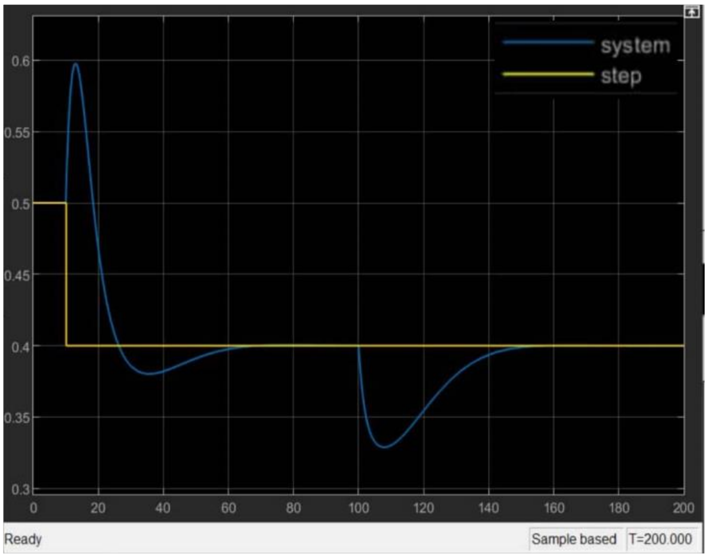


Figure 50: Variations of the Y2 Methanol mole % in Bottom and the reference as a function of time

✓ **Interpretation:**

From Figure 49 and 50 we note that the type 1 Fuzzy logic system quickly to disturbances and provided more accurate than the previous system

IV.2.3. Application of fuzzy logic control type-2:

Here we have added fuzzy logic type 2 to the system. System modeling using Simulink in MATLAB. and we obtained figure 51.

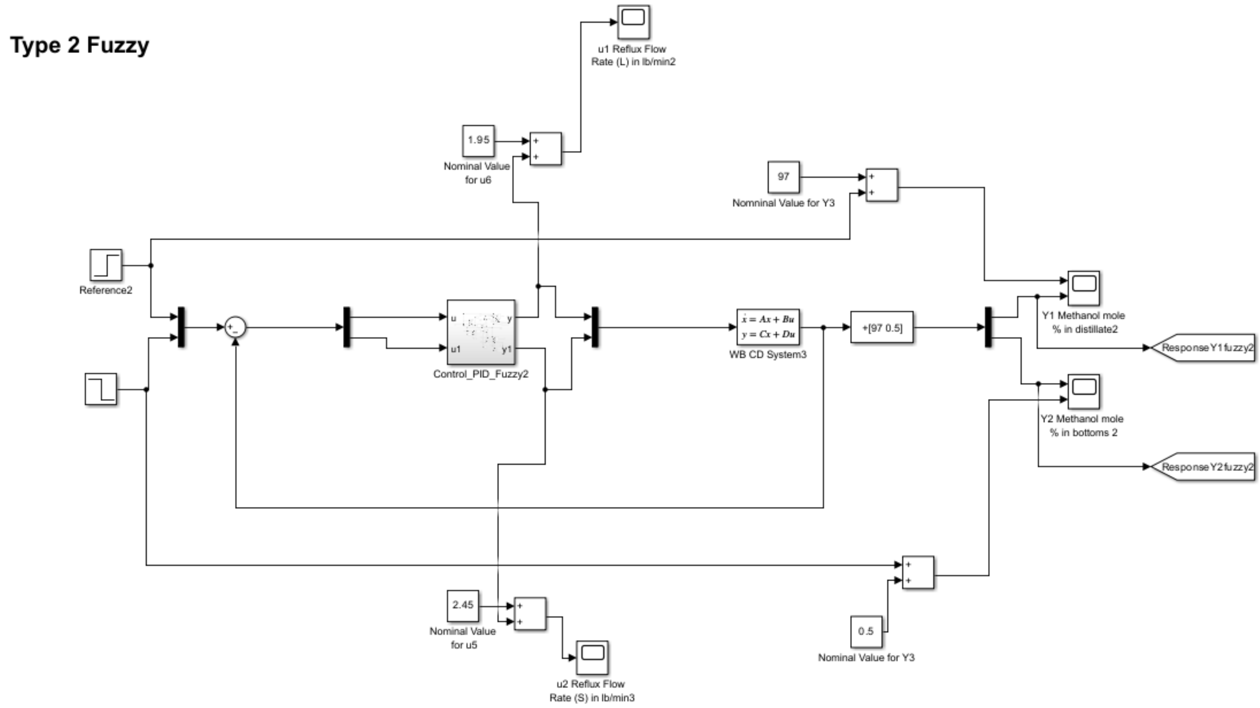


Figure 51: Block diagram of Fuzzy logic type 2 applied to column wood and berry

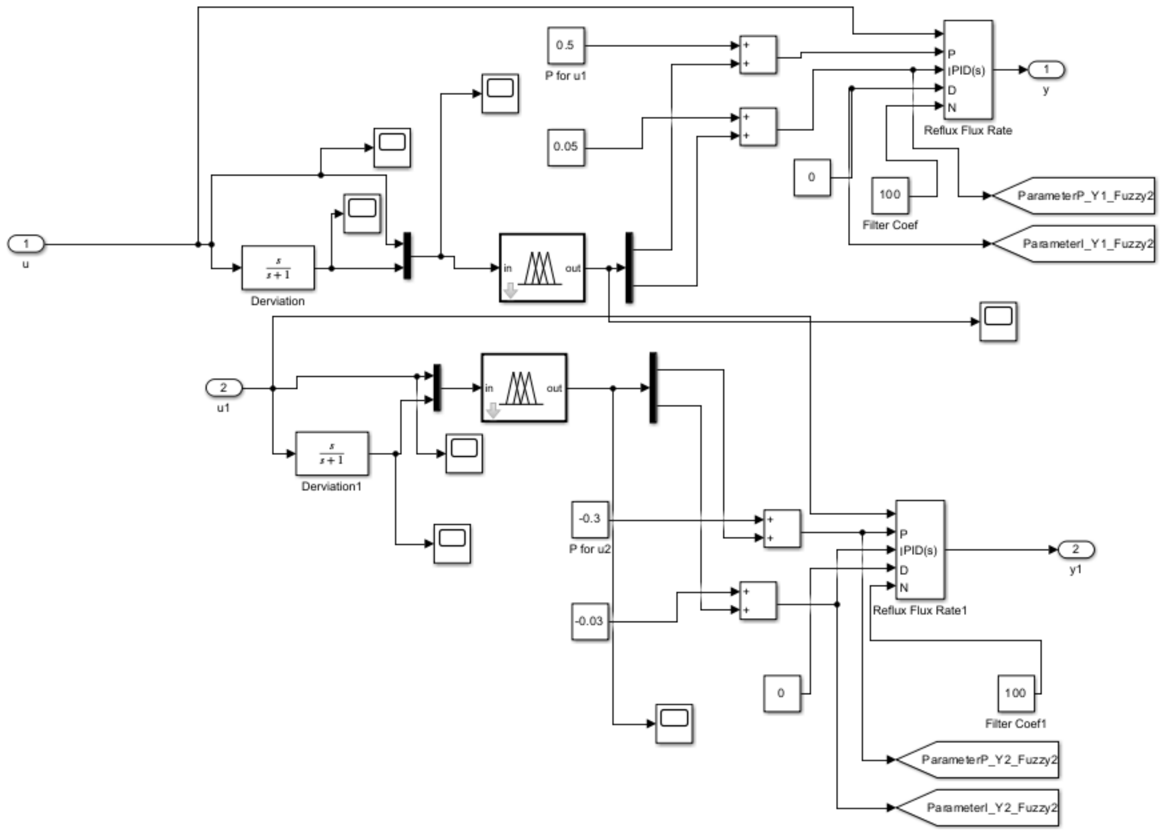


Figure 52: Sub-system of block diagram the Fuzzy PID

Fuzzy Logic Toolbox Graphical Interface:

To control this system, we will design a fuzzy MIMO controller. The " fuzzyLogicDesigner" command opens the FIS editor graphical interface, where the entire fuzzy system can be defined. By default, the interface proposes inputs and outputs using the Mamdani method. The (AND) and (OR) operations are performed using (min) and (max), respectively, inference is performed using min, rule clustering is performed using max, and defuzzification is performed using the center of gravity (center) method.

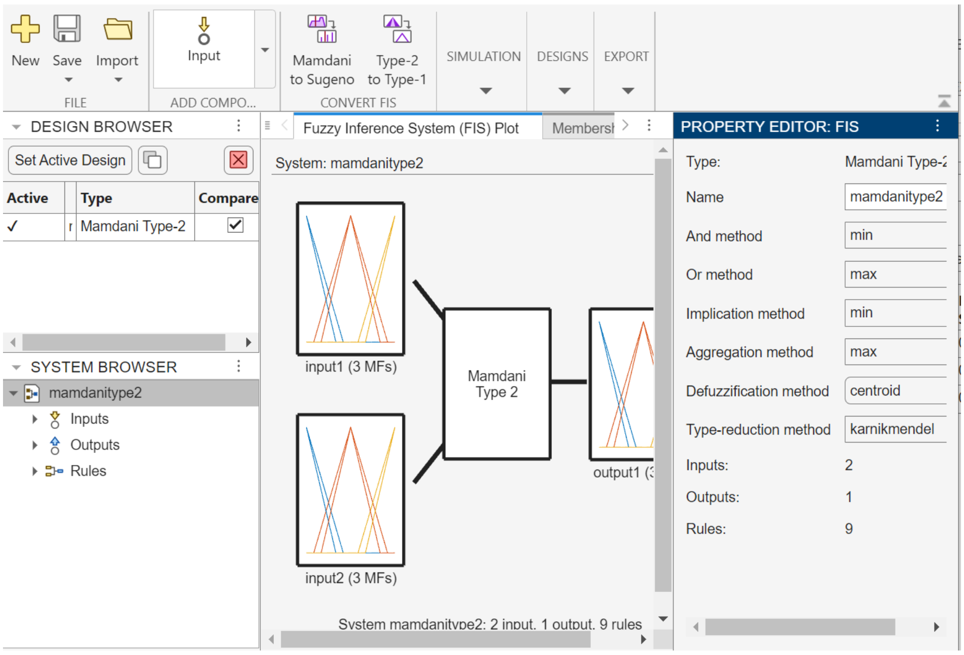


Figure 53: Graphical interface FIS Editor Fuzzy *type 2*

Fuzzy2_PID_y1:

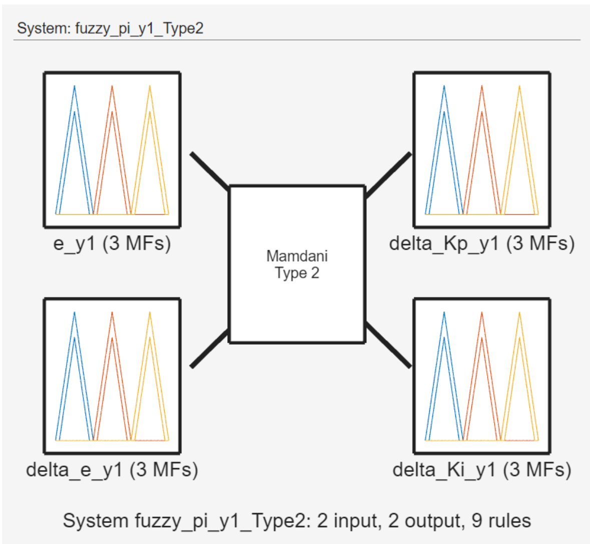


Figure 54: Graphical interface FIS Editor Fuzzy type2 y1

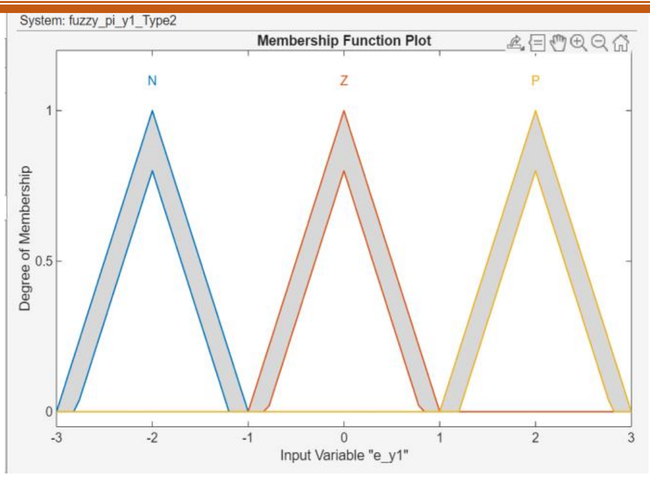


Figure 55: 1st input MFS type-2

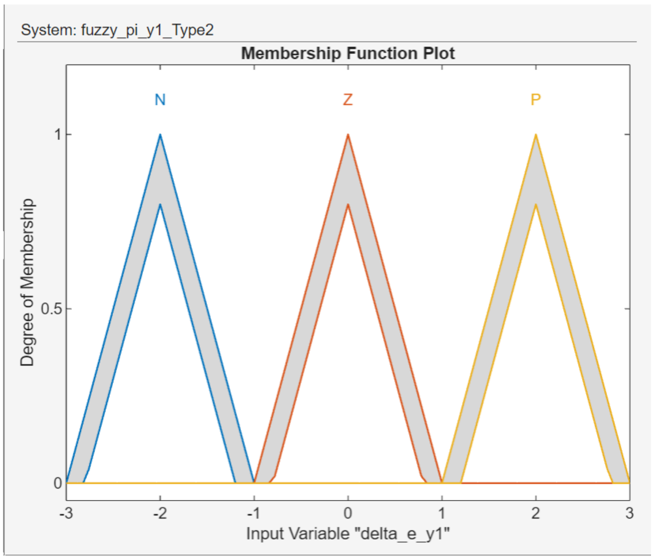


Figure 56: 2nd input MFS type-2

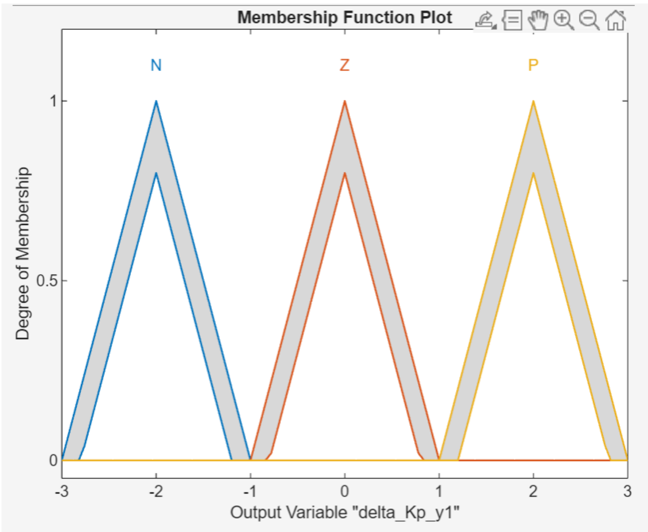


Figure 57: 1st Output MFS type-2 (Kp)

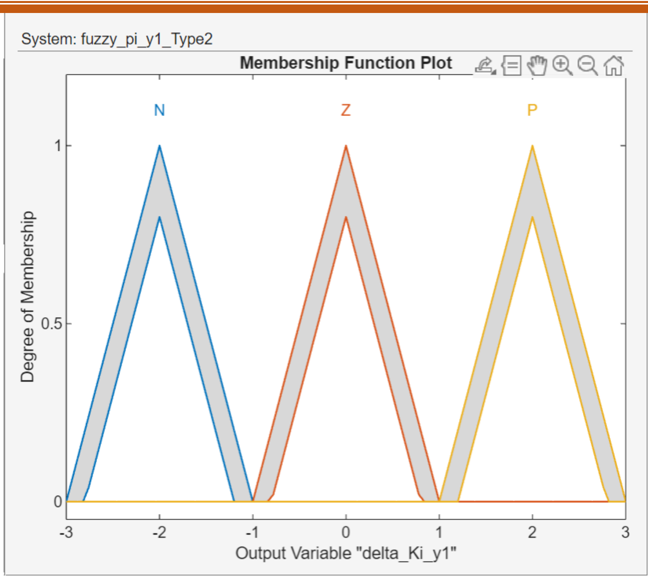


Figure 58: 3rd Output MFS type-2 (Ki)

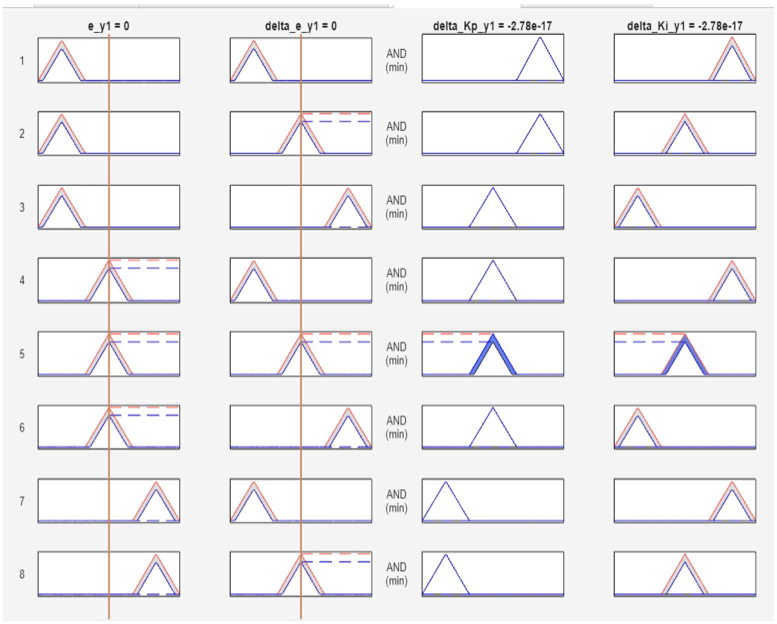


Figure 59: Rules view type 2

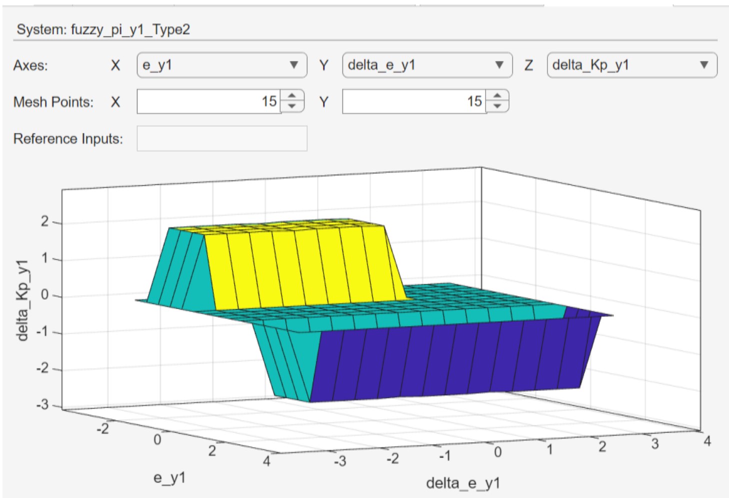


Figure 60: Surface view type-2

Fuzzy2_PID_y2:

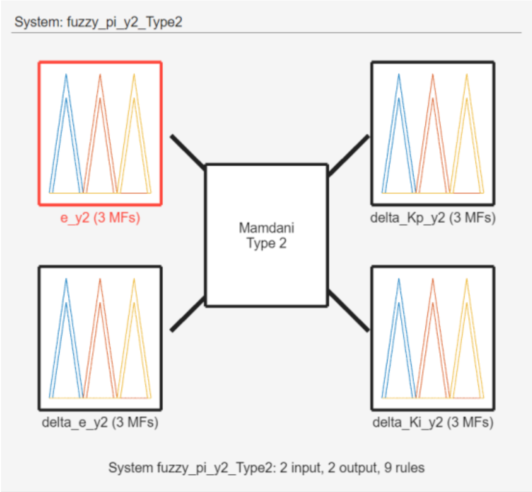


Figure 61: Graphical interface FIS Editor Fuzzy type2 y2

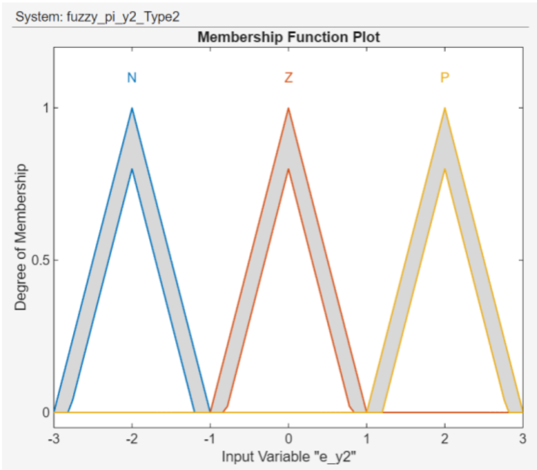


Figure 62: 1st input MFS type-2

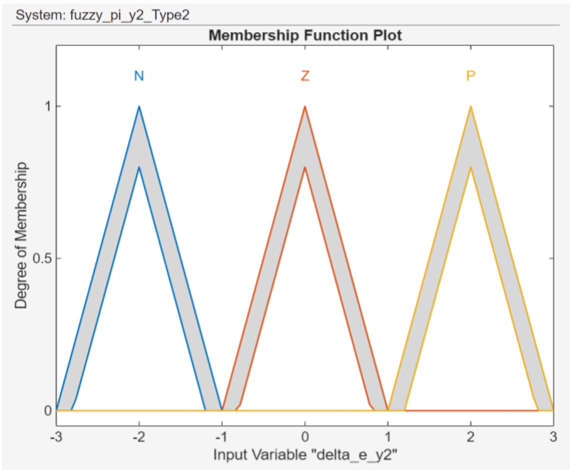


Figure 63: 2nd input MFS type-2

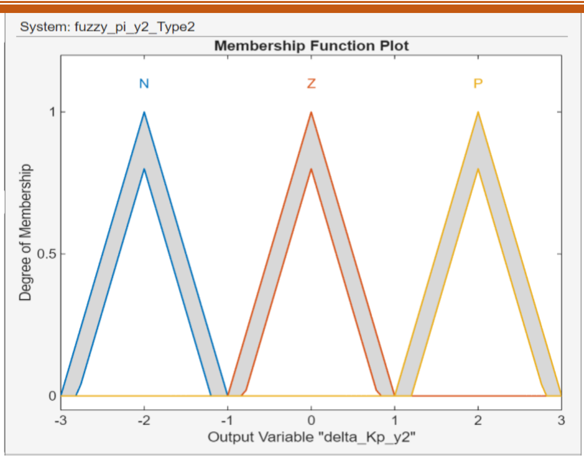


Figure 64: 1st Output MFS type-2 (Kp)

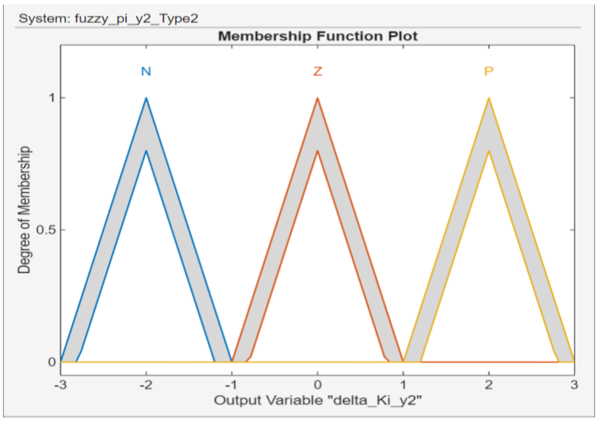


Figure 65: 3rd Output MFS type-2 (Ki)

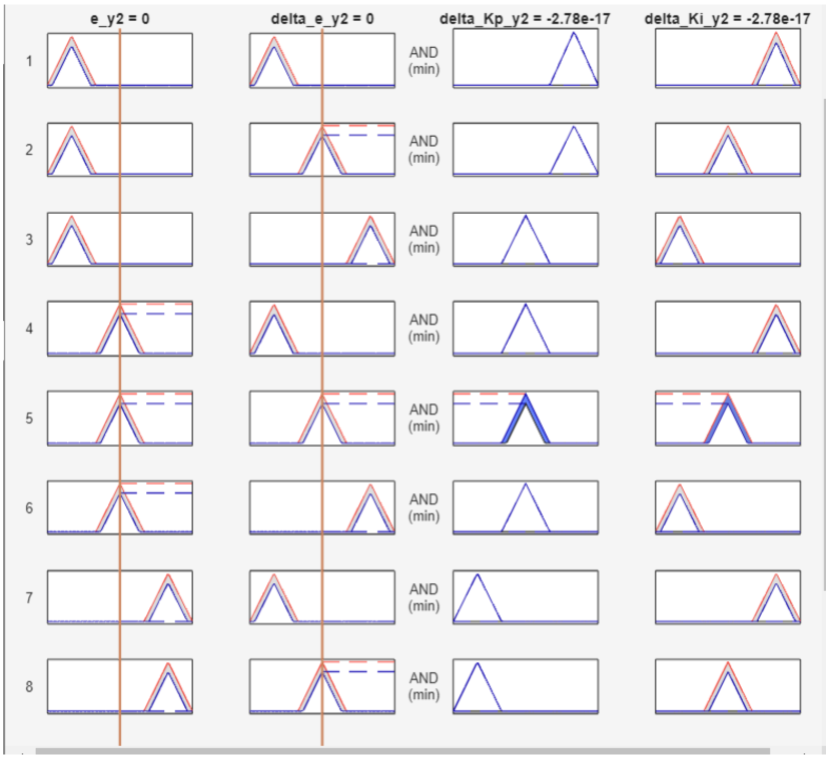


Figure 66: Rules view type 2

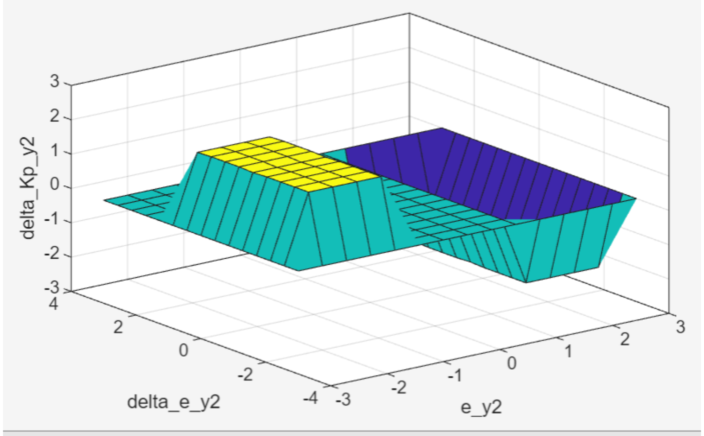


Figure 67: Surface view type-2

- **Simulink result:**

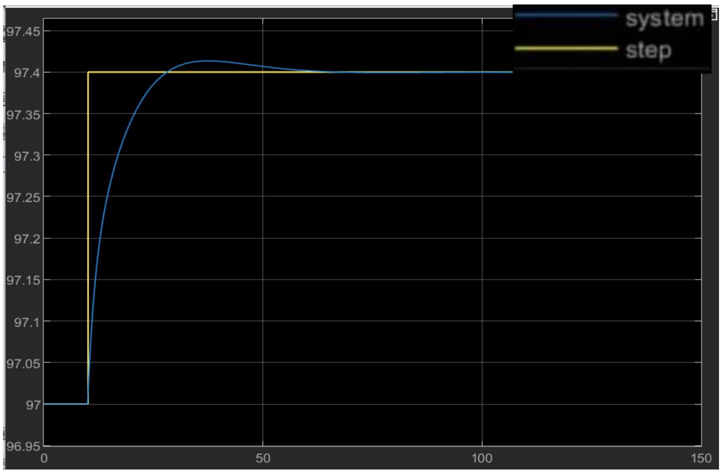


Figure 68: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

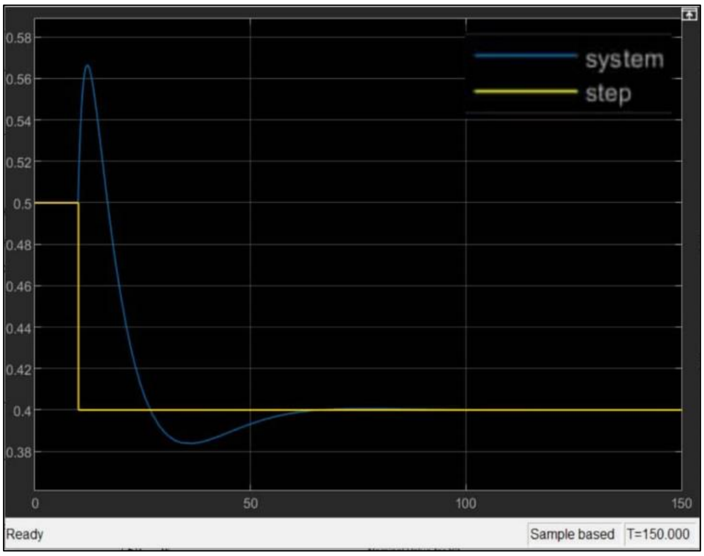


Figure 69: Variations of the Y2 Methanol mole % in Bottom and the reference as a function of time

✓ **Interpretation:**

The figure shows a very fast response to the set point with more precision and stability. The experimental results show that a better control in terms of robustness can be achieved by type-2 fuzzy logic controllersbetterse of perturbation:

- **Simulink result:**

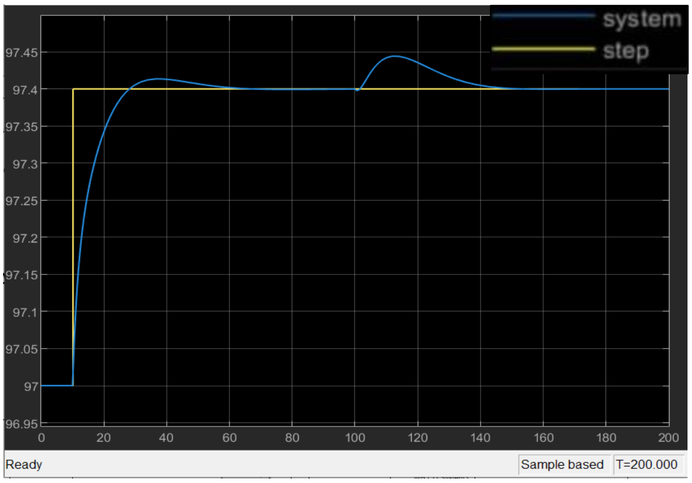


Figure 70: Variations of the Y1 Methanol mole % in distillate and the reference as a function of time

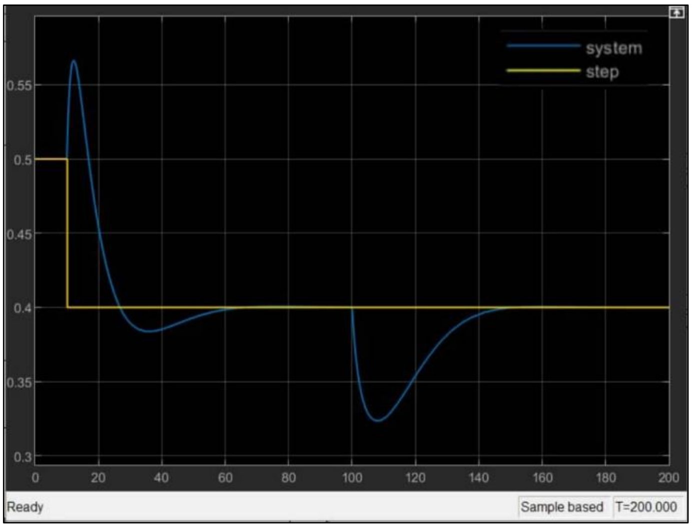


Figure 71: Variations of the Y2 Methanol mole % in Bottom and the reference as a function of time

✓ **Interpretation:**

From figure 70 and 71, we notice that the system reacted quickly to the perturbation and processed it more accurately than other System

IV.3. CONCLISION:

In this chapter, we have presented the results of simulation with the two types of fuzzy control and PID regulator which we have studied in this work, using a simple PID controller and also fuzzy

PID controller for the column wood and berry.

We have seen that the type-2 fuzzy controller shows very good performance when compared with the type-1 and PID

From the results obtained we can say that the type-2 fuzzy controller is not just an extension, but it is an improvement of the type-1 fuzzy controller in the simulation framework.

From the previous results, we see that fuzzy controllers are better than PID controllers in terms of response speed and accurcy

GENERAL CONCLUSION

GENERAL CONCLUSION

Fuzzy logic is a powerful approach for addressing a wide range of computational problems worldwide. This method has been implemented in various machines and applications to control actions according to specific pre-defined conditions. In the future, fuzzy logic is expected to be integrated into an even broader range of products and systems. As technology continues to advance and digital transformation expands, the scope of fuzzy logic applications will grow further.

The research presented in this thesis seeks to define and compare two types of control methods: fuzzy logic control and third typer controler PID control. It demonstrates that both approaches can be applied to regulate different types of systems.

The first part provides a theoretical study, presenting all the concepts and definitions required for these control methods, in order to support the development of the control structures described in this thesis.

In the second part of our work, specifically within the simulation section, we connected PID controllers as well as type-1 and type-2 fuzzy controllers to the Wood and Berry binary distillation column model. The objective was to evaluate which controller offered higher precision, faster response, and better resistance to disturbances.

Finally, we presented the simulation results for the systems controlled by the three types of controllers described earlier. The findings demonstrated that type-2 fuzzy controllers outperformed both type-1 fuzzy controllers and PID controllers in terms of overall performance.

Suggestions for Future Work

- Applying to more complex systems: It would be valuable to apply type-2 fuzzy controllers to nonlinear, multivariable, or high-order systems to further confirm their advantages over type-1 fuzzy controllers and PID controllers.
- Moving toward practical implementation: After simulation, testing the type-2 fuzzy controller on a real distillation column or a laboratory-scale experimental setup would help assess its real-world performance and reliability.
- Exploring hybrid strategies: Combining type-2 fuzzy logic with other intelligent methods, such as neural networks or genetic algorithms, could lead to hybrid controllers with even higher accuracy and stronger robustness.
- Improving energy efficiency: Another promising direction is to study how type-2 fuzzy controllers might contribute to reducing energy consumption, especially in industrial

processes like distillation where energy costs are high.

- Developing adaptive control: Research could focus on adaptive type-2 fuzzy controllers that automatically tune their parameters to cope with changing process conditions or unexpected disturbances.
- Testing robustness in challenging situations: Further experiments could examine how type-2 fuzzy controllers perform under severe disturbances, sensor failures, or actuator malfunctions, to test their robustness limits.

Broadening comparisons: Finally, it would be interesting to compare type-2 fuzzy controllers with more advanced modern control methods, such as model predictive control (MPC), sliding mode control, or reinforcement learning approaches.

BIBLIOGRAPHICAL REFERENCES

[1] Kaufman, 1973 «Introduction à la théorie des sous-ensembles flous», Tome1, Masson, Paris.

[2] L. Gacogne, 1997 «Elément de logique floue», Hermes.

[3] P. Prokopowicz, J. Czerniak, D. Mikolajewski, L. Apiecionek, D. Slezak, 2017

[4] «Theory and application of ordered fuzzy numbers», Polish Academy of Sciences, Warsaw, Poland.

[5] Y. M. Nafrizuan, E. Nurainaa, M. H. Mohd Nordin, 2022«A comparison of type 1 and type 2 fuzzy logic controller for DC motor system», University Malaysia, Pahang, Gambang, Malaysia.

[6] M. Gr. Voskoglou, 2019 «Fuzzy Sets, Fuzzy Logic and Their Applications», Editorial Office MDPI St. Alban-Anlage 66, 4052 Basel, Switzerland**.**

[7] Y. Bai, D. wang, 2006 « Fundamentals of Fuzzy Logic Control – Fuzzy Sets, Fuzzy Rules and Defuzzifications », Johnson C.smith University.

[8] The history of fuzzy logic, [online], available on; https://www.computerworld.com/article/2566110/sidebar--the-history-of-fuzzy-logic.html Consulted on: 17/04/2022

[9] A. KHEMIS, 10/04/2011 « Application de la logique floue type-2 intervalle à la commande adaptative de la machine asynchrone», Magister Thesis in Electrotechnique, électricité industrielle, University Mohamed Khider, Biskra.

[10] **F. Dernoncourt**, January 2013 «introduction to fuzzy logic».

[11] **R. Bendib**, 2021/2022 «Course Artificial intelligence», Master 2 auto, University 20 auguest 1955, Skikda,.

[12] **T. J. Ross,** 2010 «Fuzzy Logic with Engineering Applications», 3rd edition, University of New Mexico, USA, John Wiley & Sons, Ltd.

[13] **Ion,** 2012 «A Mamdani Type Fuzzy Logic Controller. Fuzzy Logic: Controls, Concepts, Theories and Applications, pp. 325-350.

[14] **N. N. Karnik, J. M. Mendel, Q. Liang**, December 1999 «Type-2 fuzzy logic systems», IEEE transaction on fuzzy systems, vol. 7, No. 6.

[15] **N. N. Karnik, J. M. Mendel,** 2001 «Operations on type-2 fuzzy sets », Fuzzy Sets and Systems 122, pp. 327–348.

A. S. Shvedov, December 2021 **«** On type-2 fuzzy sets and type-2 fuzzy systems **»,**

[16] Journal of Mathematical Sciences, Vol. 259, No. 3.

[17] **S. Bezzaoucha,** 2007 «Différentes stratégies de commande floue appliquées aune colonne d'absorption», Thesis prepared at the Process Control Laboratory 10, Avenue Hassen Badi, BP 182, EL-Harrach, Alger.

[18] **M. Benbrahim, N. Essounbouli, A. Hamzaoui, A. Betta,** August 2013 « Adaptive Type-2 Fuzzy Sliding Mode Controller for SISO Nonlinear Systems Subject to Actuator Faults»,International Journal of Automation and Computing 10(4), 335-342.

[19] **Fuzzy Logic Tutorial: What is, Architecture, Application, Example,** [online], available on; https://www.guru99.com/what-is-fuzzy-logic.html, consulted on: 01/06/2022.

[20] **Fuzzy logic, robotics and future of artificial intelligence,** [online], available on; https://www.softwaretestinghelp.com/fuzzy-logic-robotics-in ai/#Fuzzy_Logic_In_Artificial_Intelligence , consulted on : 04/05/2022.

[21] **Fuzzy logic,** [online], available on; https://www.techtarget.com/searchenterpriseai/definition/fuzzy-logic , consulted on: 27/05/2022.

[22] (Seader, J. D., Henley, E. J., & Roper, D. K. (2011). Separation Process Principles (3rd ed.). Wiley.).

[23] (Luyben, W. L. (2006). Distillation Design and Control Using Aspen Simulation. Wiley-Interscience, Hoboken, NJ, USA, pp. 1–40. ISBN: 978-0471778882)

[24] (Seader, J. D., Henley, E. J., & Roper, D. K. (2011). Separation Process Principles (3rd ed.). Wiley, New York, USA, pp. 237–239. ISBN: 978-0470646113

[25] (Skogestad, S., & Postlethwaite, I. (2007). Multivariable Feedback Control: Analysis and Design (2nd ed.). Wiley, West Sussex, UK, pp. 5–9. ISBN: 978-0470011683).

[26] (Ogunnaike, B. A., & Ray, W. H. (1994). Process Dynamics, Modeling, and Control. Oxford University Press, New York, USA, pp. 487–495. ISBN: 978-0195091199).

[27] (Seborg, D. E., Edgar, T. F., Mellichamp, D. A., & Doyle, F. J. (2010). Process Dynamics and Control (3rd ed.). Hoboken, NJ: John Wiley & Sons, pp. 420–422. ISBN: 978-0470128671.).

[28] (Wood, R. K., & Berry, M. W. (1973). "Terminal Composition Control of a Binary Distillation Column." Chemical Engineering Science, 28(9), pp. 1707–1717. DOI: 10.1016/0009-2509(73)85027-6).

[29] (Skogestad, S., & Postlethwaite, I. (2007). Multivariable Feedback Control: Analysis and Design (2nd ed.). Wiley, West Sussex, UK, pp. 5–9. ISBN: 978-0470011683).

[30] (Wood, R. K., & Berry, M. W. (1973). "Terminal Composition Control of a Binary Distillation Column." Chemical Engineering Science, 28(9), pp. 1708. DOI: 10.1016/0009-2509(73)85027-6).

[31] (Luyben, W. L. (2006). Distillation Design and Control Using Aspen Simulation. Wiley-Interscience, Hoboken, NJ, USA, pp. 36–40. ISBN: 978-0471778882)

[32] (Skogestad, S., & Postlethwaite, I. (2007). Multivariable Feedback Control: Analysis and Design (2nd ed.). Wiley, West Sussex, UK, pp. 9. ISBN: 978-0470011683).

[33] (Mendel, J. M. (2001). Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall, Upper Saddle River, NJ, USA, pp. 60–62. ISBN: 978-0130409690).

[34] (Amir Ghoreishi Islamic Azad University Varamin (march 2011) Article in International Journal of Advancements in Computing Technology ·pp39. DOI: 10.4156/ijact.vol3.issue1.4)

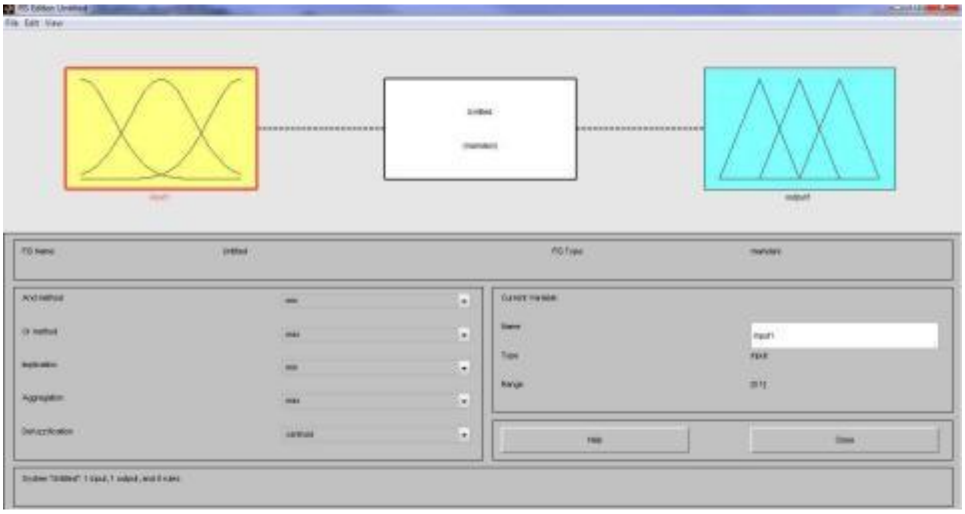
APPENDIX

Appendix No. 01

Instructions for Using the Graphical Interface of the MATLAB Fuzzy Logic Toolbox

Graphical Interface of the Fuzzy Logic Toolbox

By using the fuzzy command, you can launch the FIS Editor graphical interface, which allows you to fully design and configure the fuzzy logic system.



By default, the interface provides one input and one output using the Mamdani method. The AND and OR operators are implemented through the minimum and maximum functions, respectively. The implication uses the minimum operator, rule aggregation is handled by the maximum operator, and defuzzification is performed using the centroid (center of gravity) method.

The various menus and their available options are described in the following sections.

<u>File menue</u>

- **New FIS :**
  - Mamdani: creates a new fuzzy system of the Mamdani type.
  - Sugeno: creates a new fuzzy system of the Sugeno type.

- **Import :**
  - From workspace loads the fuzzy system matrix from the MATLAB workspace.
  - From file loads a fuzzy system previously saved on disk as a .FIS file.

- **Export:**

  - To workspace: saves the fuzzy system into the MATLAB workspace as a matrix, which is necessary if you wish to use the system in SIMULINK.
  - To file: saves the current fuzzy system to a file on disk.

Edit Menu

- **Undo:** cancels the most recent change.
- **Add Variable :**

  - Input: adds a new input variable.
  - Output: adds a new output variable.

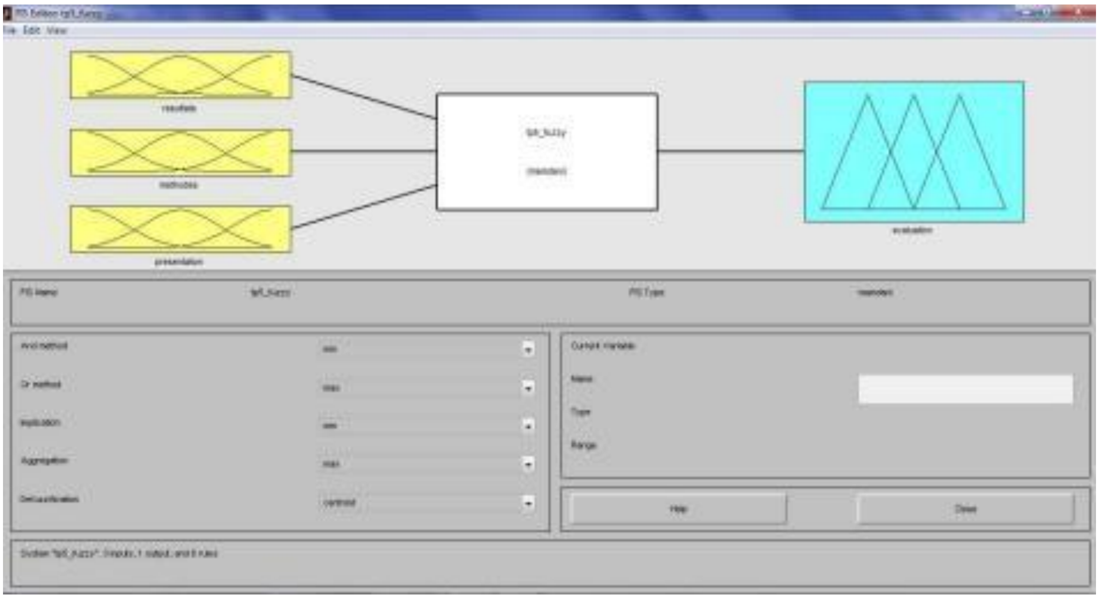
- **Remove Selected Variable:** deletes the input or output variable that was previously selected with the mouse.

View Menu

- **View Rules:** opens the Rule Viewer window, where you can assign values to all input variables either by using the mouse or by typing numerical values into the input vector. This feature lets you visualize the fuzzy output surface along with its numerical value after defuzzification.
- **View Surface:** opens the Surface Viewer window, where you can examine the output surface as a function of the input variables.
- **Edit FIS Properties:** allows you to return to the FIS Editor window, where you can modify all system properties while working in the Rule Editor or the Membership Function Editor windows.
- **Edit Membership Functions:** opens the membership function editor for the currently selected variable.

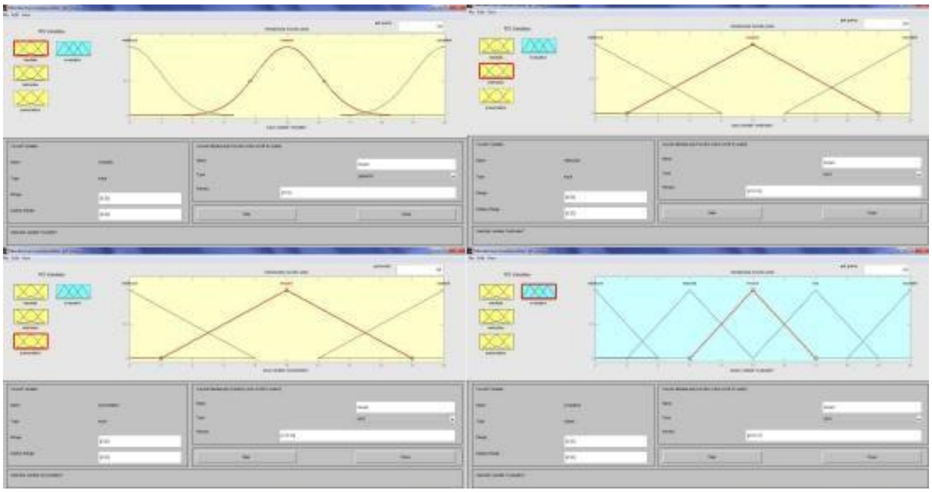
Manipulation

In the FIS Editor window (initially labeled untitled and opened using the fuzzy command), you can add additional input variables by selecting the Add Input option from the Edit menu. For each input or output variable, which you can select with the mouse, you have the option to assign a name and choose between different methods such as max-min, sum-product, and others.

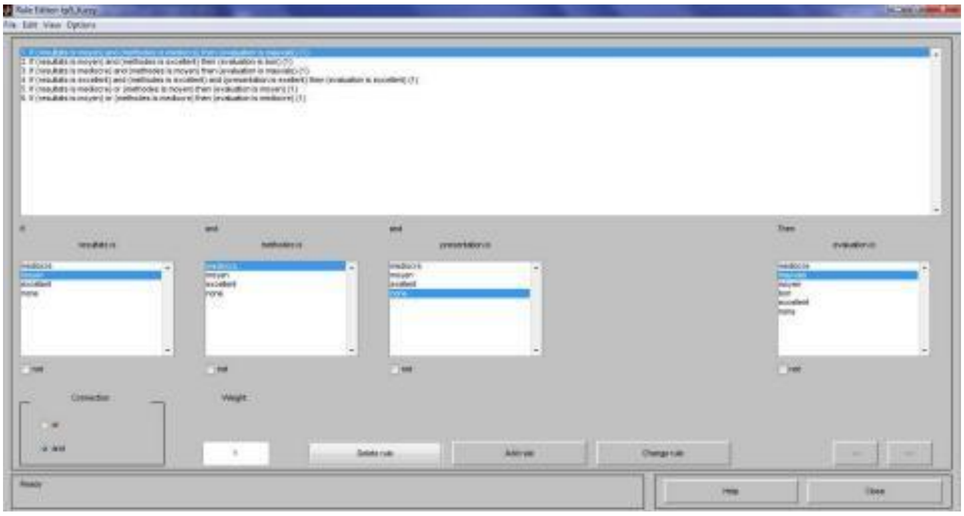


By double-clicking on any of these variables, you will open the membership function editor window, where you can select the number and type of membership functions as well as define the variable’s value range.

Each membership function can then be selected with the mouse to assign it a name, which will be referenced later in the fuzzy rules.



Once all input and output variables have been fully defined, you can open the Rule Editor window by choosing the Edit Rules option. This allows you to create and edit the fuzzy rules.



In this window, a new menu called Options lets you select the language for writing the rules. Each rule can also be assigned a weighting factor, specified at the end of the rule in brackets; by default, this weight is set to 1.

The Edit FIS Properties option in the View menu allows you to return to the FIS Editor window. The fuzzy system can then be saved as a file with the .FIS extension and a chosen name.

If you wish to use this system in the workspace, within SIMULINK, or inside an M file, it is essential to save it in the workspace as a matrix that fully describes the system, using the Export to Workspace option.

>> sys_flou= readfis('fuzzy') sys_flou =

name: 'fuzzy' type: 'mamdani' andMethod: 'min' orMethod: 'max'

defuzzMethod: 'centroid' impMethod: 'min' aggMethod: 'max'

input: [1x3 struct] output: [1x1 struct] rule: [1x6 struct].

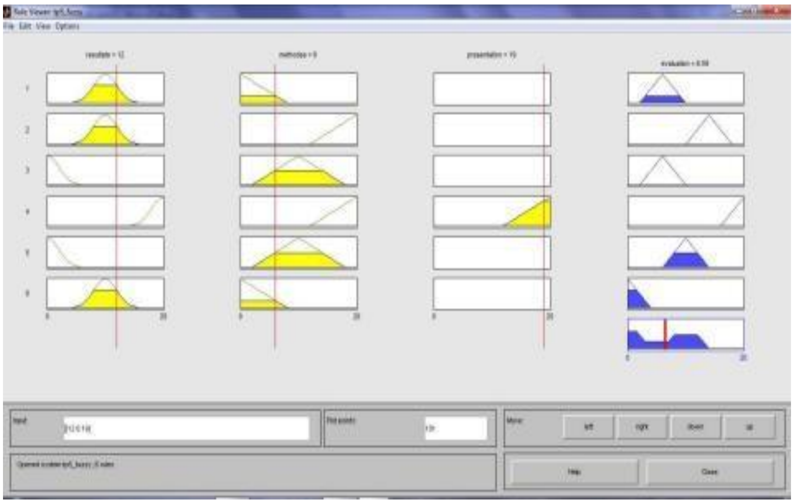
If, for example, we want to see the output of the fuzzy system in front of three given notes, we define these three notes x1, x2 and x3, and then we evaluate the result with the command 'evalfis', as follows:

>> x=[x1 x2 x3]

>> y=evalfis(x,sys_flou)

You can view the rule editor by typing the command:

>> ruleview(sys_flou)



You can modify the three values in the Input section, and you will notice that the yellow and blue membership sets will adjust accordingly with the changes in the input vector. Naturally, the output—shown by the red bar with its numerical value displayed at the top right in the conclusions area—will also update to reflect these changes.