جامعة ديالى
كلية الهندسة
قسم هندسة الحاسوب

# Fuzzy Logic Controller

By

Dr. Saad  A. Salman

# Fuzzy control system

A fuzzy control system is a control system based on fuzzy logic — a mathematical system that analyzes analog input values in terms of logical variables that take on continuous values between 0 and 1, in contrast to classical or digital logic, which operates on discrete values of either 1 or 0 (true or false, respectively).

# Overview

Fuzzy logic is widely used in machine control. The term "fuzzy" refers to the fact that the logic involved can deal with concepts that cannot be expressed as the "true" or "false" but rather as "partially true". Although alternative approaches such as genetic algorithms and neural networks can perform just as well as fuzzy logic in many cases, fuzzy logic has the advantage that the solution to the problem can be cast in terms that human operators can understand, so that their experience can be used in the design of the controller. This makes it easier to mechanize tasks that are already successfully performed by humans.

# History

Fuzzy logic was first proposed by Lotfi A. Zadeh of the University of California at Berkeley in a 1965 paper. He elaborated on his ideas in a 1973 paper that introduced the concept of "linguistic variables", which in this article equates to a variable defined as a fuzzy set. Other research followed, with the first industrial application, a cement kiln built in Denmark, coming on line in 1975.

# Fuzzy sets

The input variables in a fuzzy control system are in general mapped by sets of membership functions similar to this, known as "fuzzy sets". The process of converting a crisp input value to a fuzzy value is called "fuzzification".

A control system may also have various types of switch, or "ON-OFF", inputs along with its analog inputs, and such switch inputs of course will always have a truth value equal to either 1 or 0, but the scheme can deal with them as simplified fuzzy functions that happen to be either one value or another.

# Fuzzy control

Fuzzy controllers are very simple conceptually. They consist of an ***input stage***, a ***processing stage***, and an ***output stage***. The input stage maps sensor or other inputs, such as switches, thumbwheels, and so on, to the appropriate membership functions and truth values. The processing stage invokes each appropriate rule and generates a result for each, then combines the results of the rules. Finally, the output stage converts the combined result back into a specific control output value.

Most common shape of membership functions is triangular, although trapezoidal and bell curves are also used, but the shape is generally less important than the number of curves and their placement. From three to seven curves are generally appropriate to cover the required range of an input value.

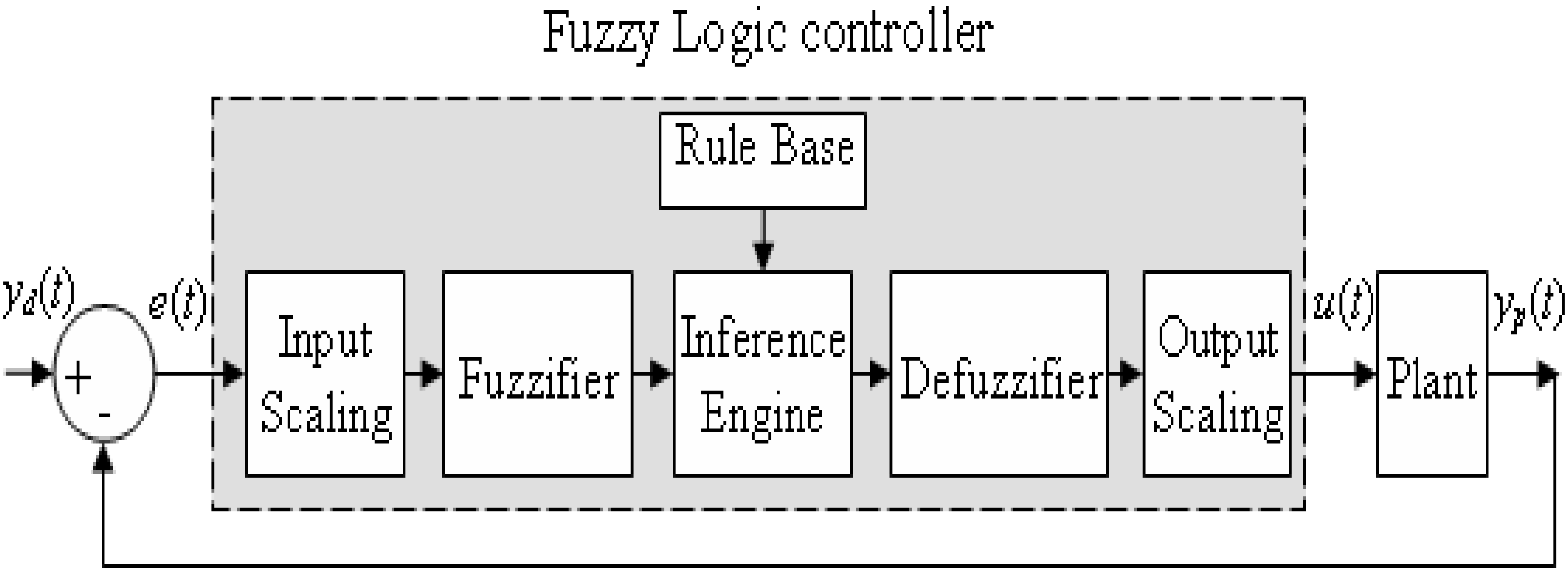
The processing stage is based on a collection of logic rules in the form of ***IF-THEN*** statements, where the IF part is called the "antecedent" and the THEN part is called the "consequent". Typical fuzzy control systems have dozens of rules.

In practice, the fuzzy rule sets usually have several antecedents that are combined using fuzzy operators, such as AND, OR, and NOT, though again the definitions tend to vary: AND, in one popular definition, simply uses the minimum weight of all the antecedents, while OR uses the maximum value. There is also a NOT operator that subtracts a membership function from 1 to give the "complementary" function.

Rules can be solved in parallel in hardware, or sequentially in software. The results of all the rules that have fired are "defuzzified" to a crisp value by one of several methods.

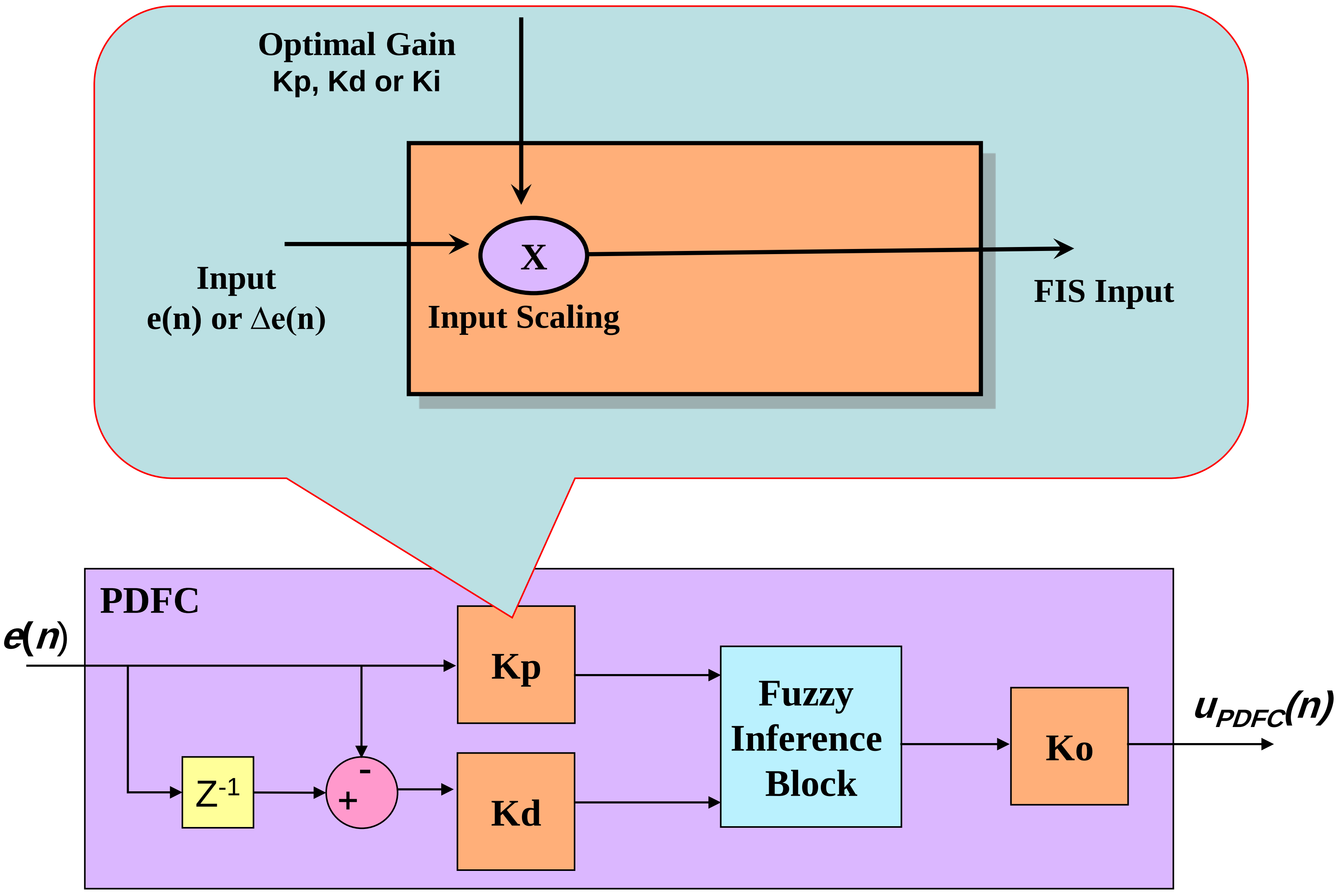
# Fuzzy Logic Controller Design

Fuzzy Logic has been successfully applied to a large number of control applications. The most commonly used controller is the PID controller, which requires a mathematical model of the system. A fuzzy logic controller provides an alternative to the PID controller. It is a good tool for the control of systems that are difficult to model fuzzy logic has rapidly become one of the most successful of today's technologies for developing sophisticated control systems. With its aid, complex requirements may be implemented in amazingly simple, easily maintained, and inexpensive controllers, where the Structure of the typical fuzzy logic controller shown in Figure below:

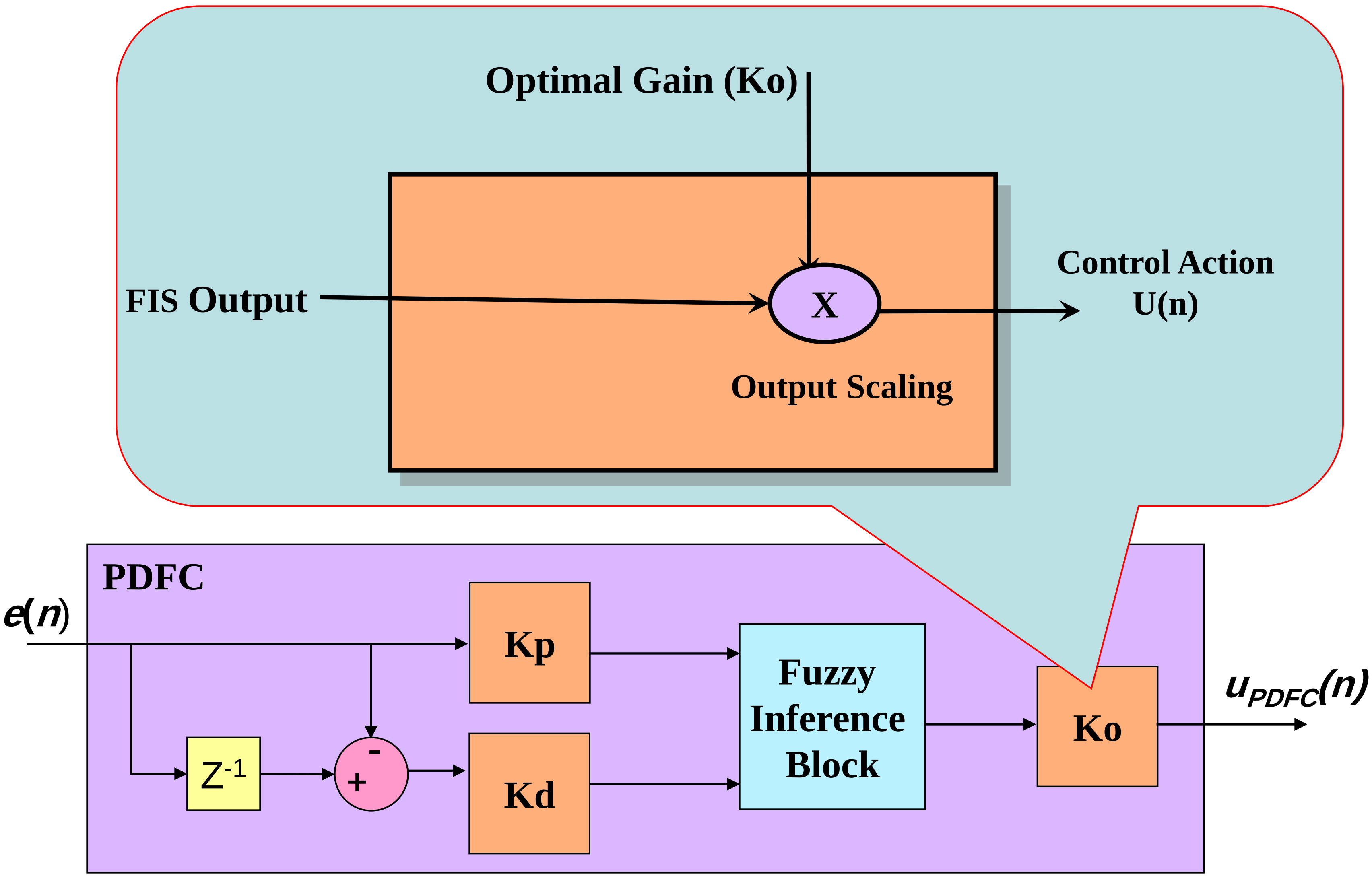



**Structure of Fuzzy Logic Controller with Unity Feedback Control System**

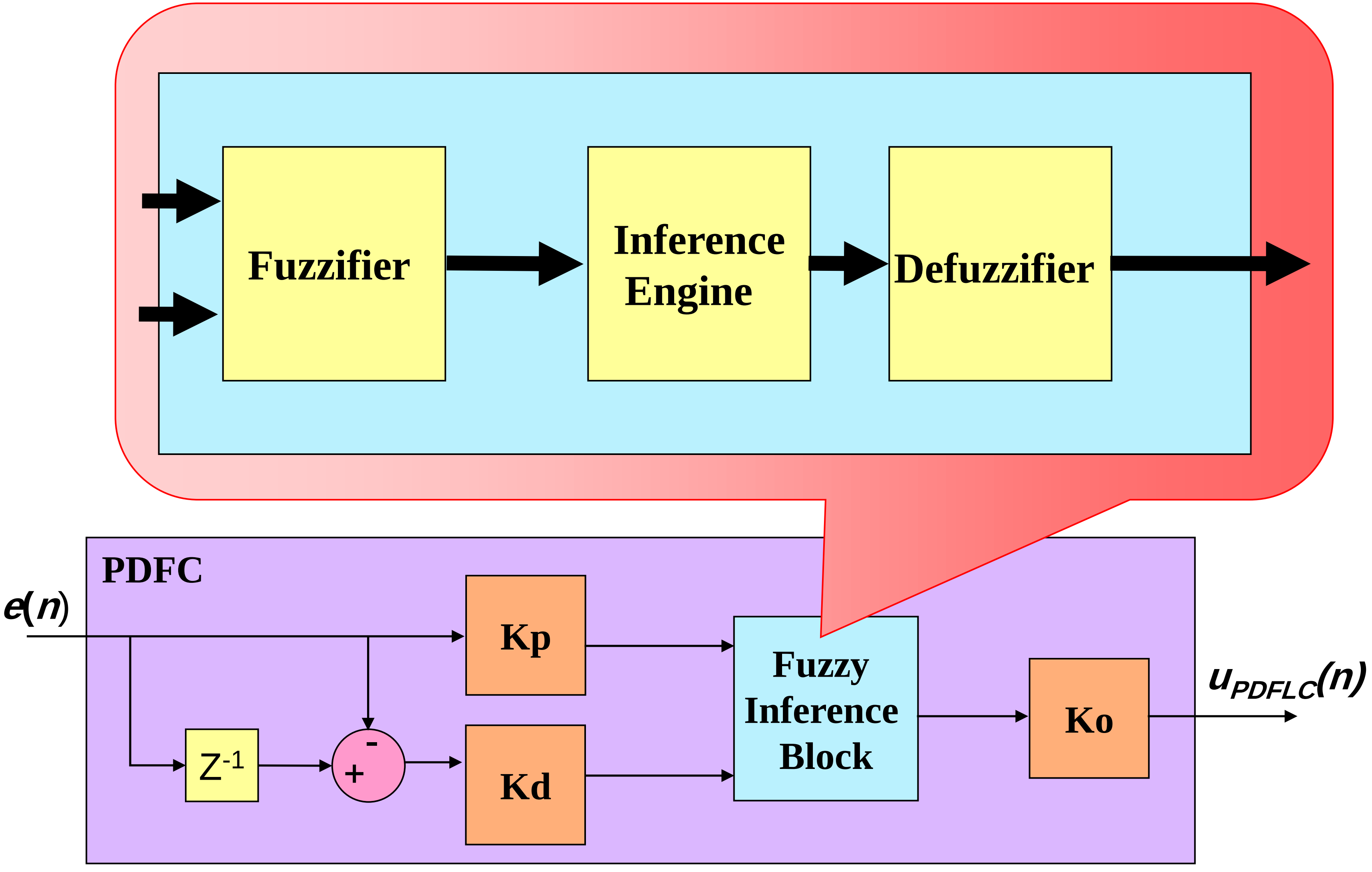
# Inputs Tuning Gain block



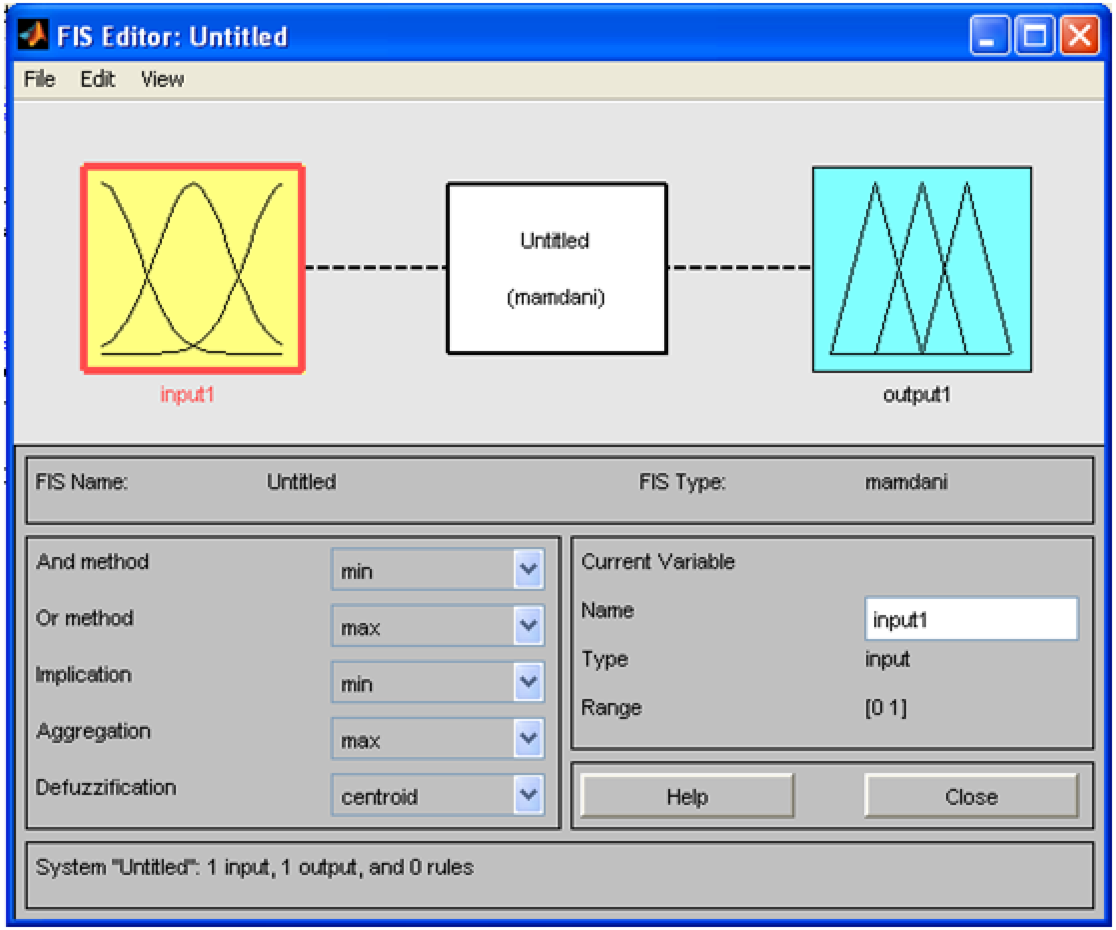
# Output Tuning Gain block



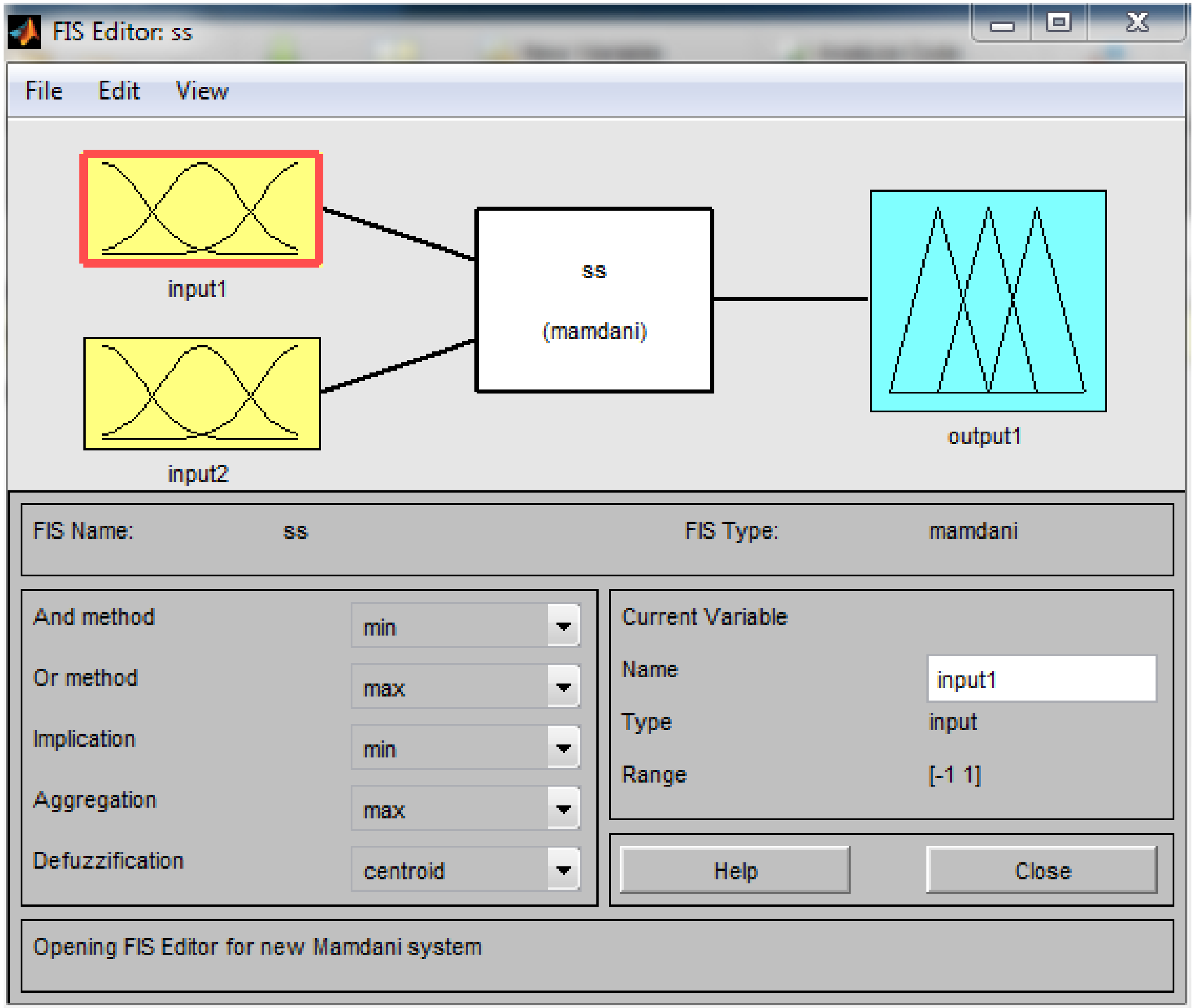
# PDFC structure with FIS

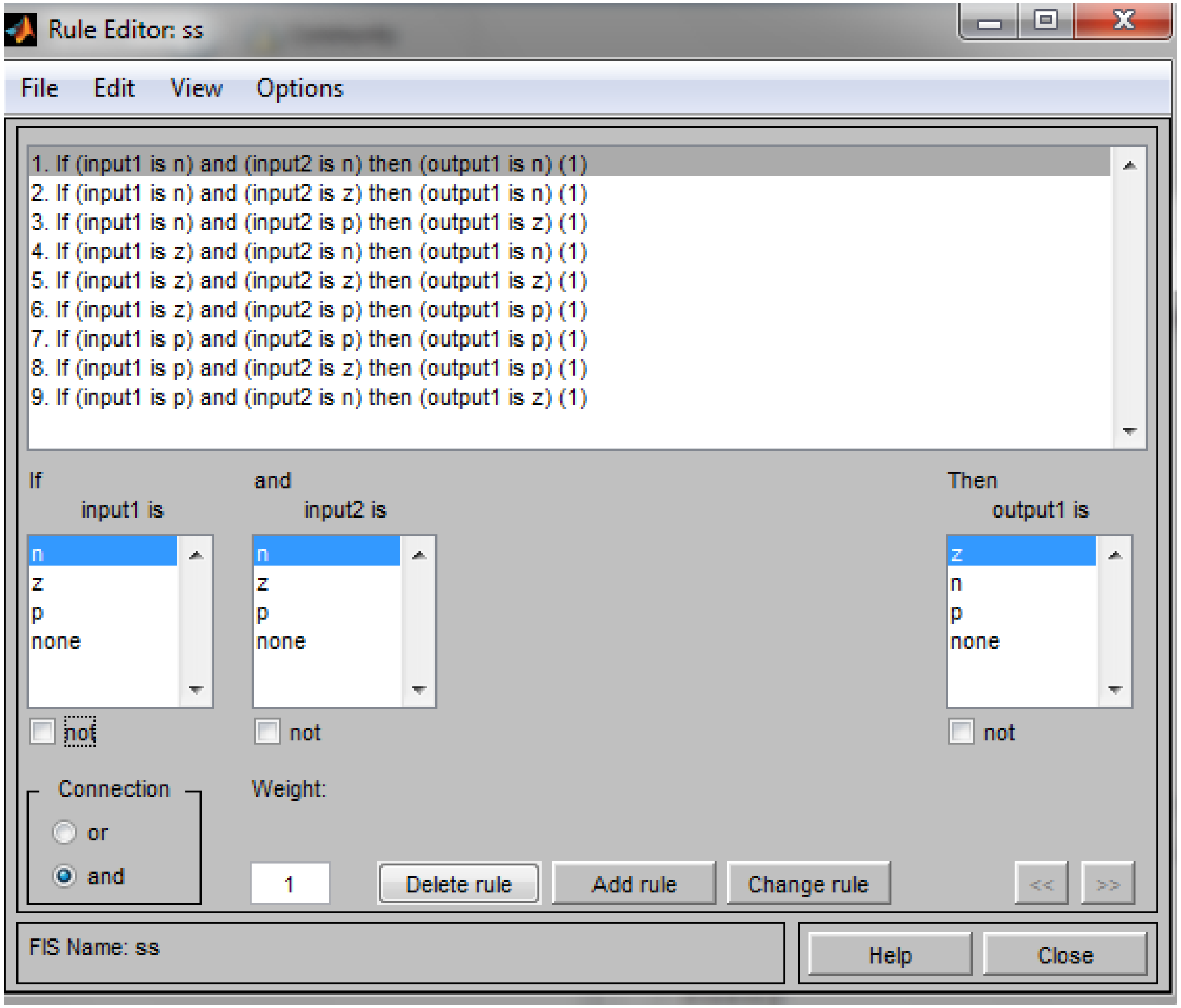


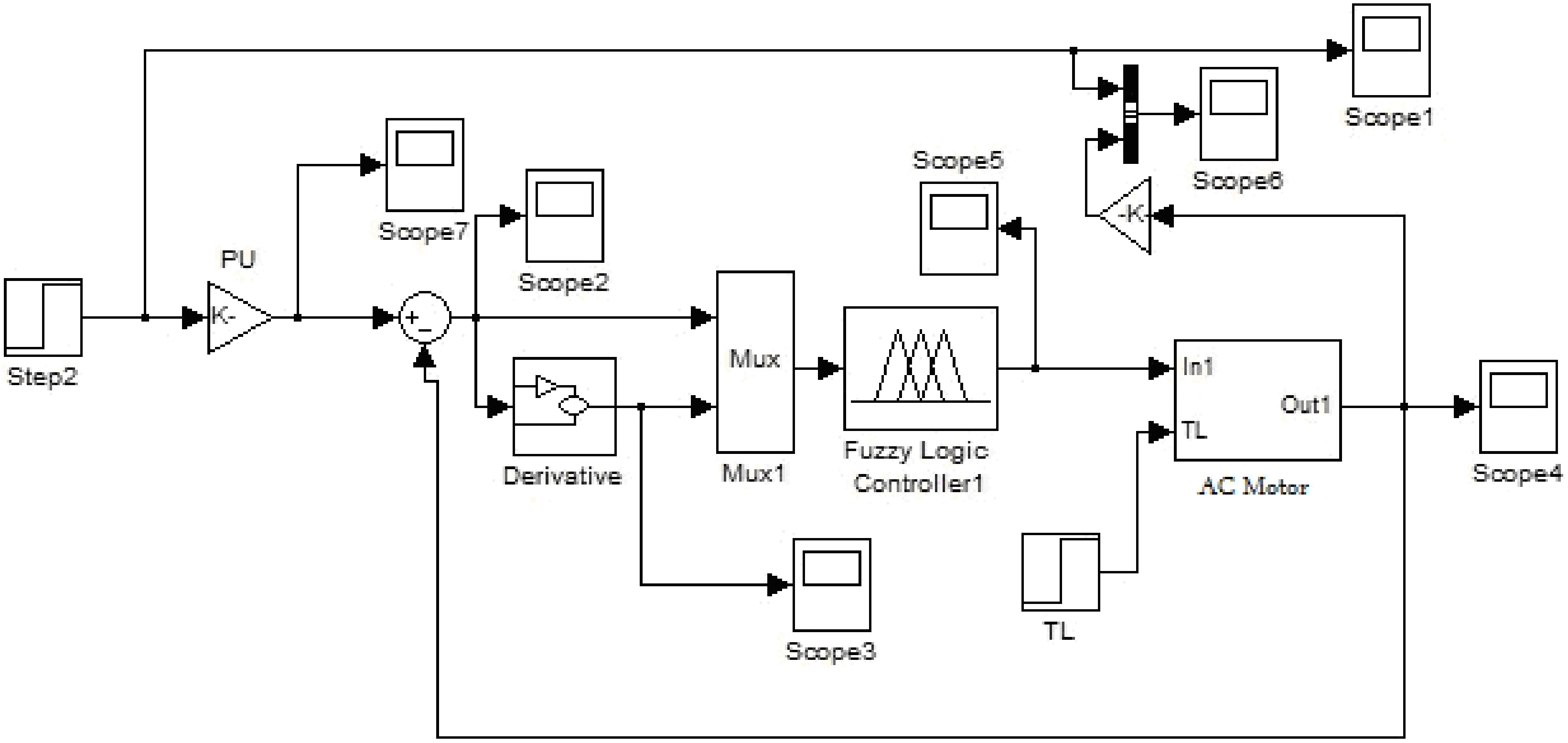
The control action in fuzzy logic controllers can be expressed with membership function and simple “if-then”.




**Typical Fuzzy Logic Toolbox**


FIS Editor: ss
File
Edit
View
input1
input2
ss
(mamdani)
output1
FIS Name:
ss
FIS Type:
mamdani
And method
min
Or method
max
Implication
min
Aggregation
max
Defuzzification
centroid
Current Variable
Name
input1
Type
input
Range
[-1 1]
Help
Close
Opening FIS Editor for new Mamdani system


Rule Editor: ss
File Edit View Options
1. If (input1 is n) and (input2 is n) then (output1 is n) (1)
2. If (input1 is n) and (input2 is z) then (output1 is n) (1)
3. If (input1 is n) and (input2 is p) then (output1 is z) (1)
4. If (input1 is z) and (input2 is n) then (output1 is n) (1)
5. If (input1 is z) and (input2 is z) then (output1 is z) (1)
6. If (input1 is z) and (input2 is p) then (output1 is p) (1)
7. If (input1 is p) and (input2 is p) then (output1 is p) (1)
8. If (input1 is p) and (input2 is z) then (output1 is p) (1)
9. If (input1 is p) and (input2 is n) then (output1 is z) (1)
If
input1 is
n
z
p
none
not
and
input2 is
n
z
p
none
not
Then
output1 is
z
n
p
none
not
Connection
or
and
Weight:
1
Delete rule
Add rule
Change rule
FIS Name: ss
Help
Close



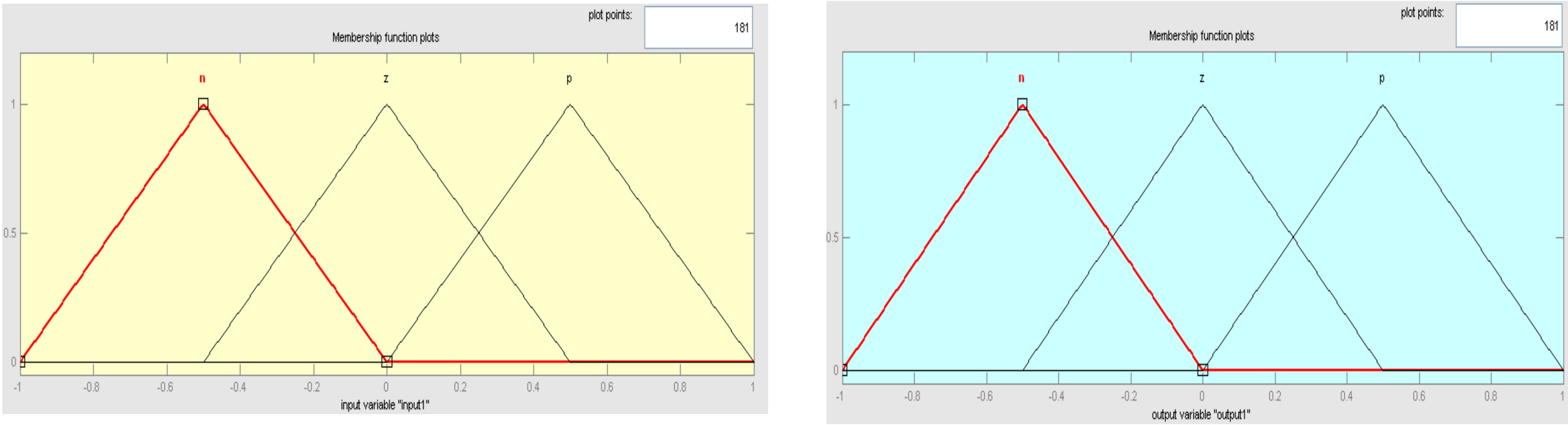

**Matlab Simulink of fuzzy controller with the position control of an AC motor**

# Results and Discussion

## 1. Membership Function of 3*3 Rule Base

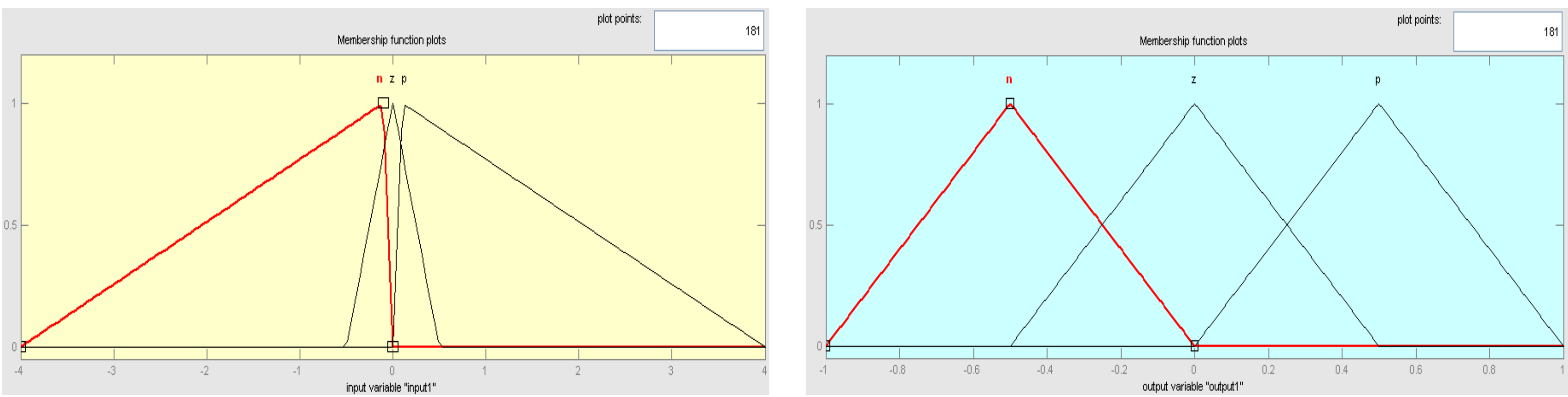
| e/de | N | Z | P |
|---|---|---|---|
| N | N | Z | Z |
| Z | Z | Z | Z |
| p | Z | Z | P |

Fuzzy rule base of 9 rules for 3*3 two inputs fuzzy Pd



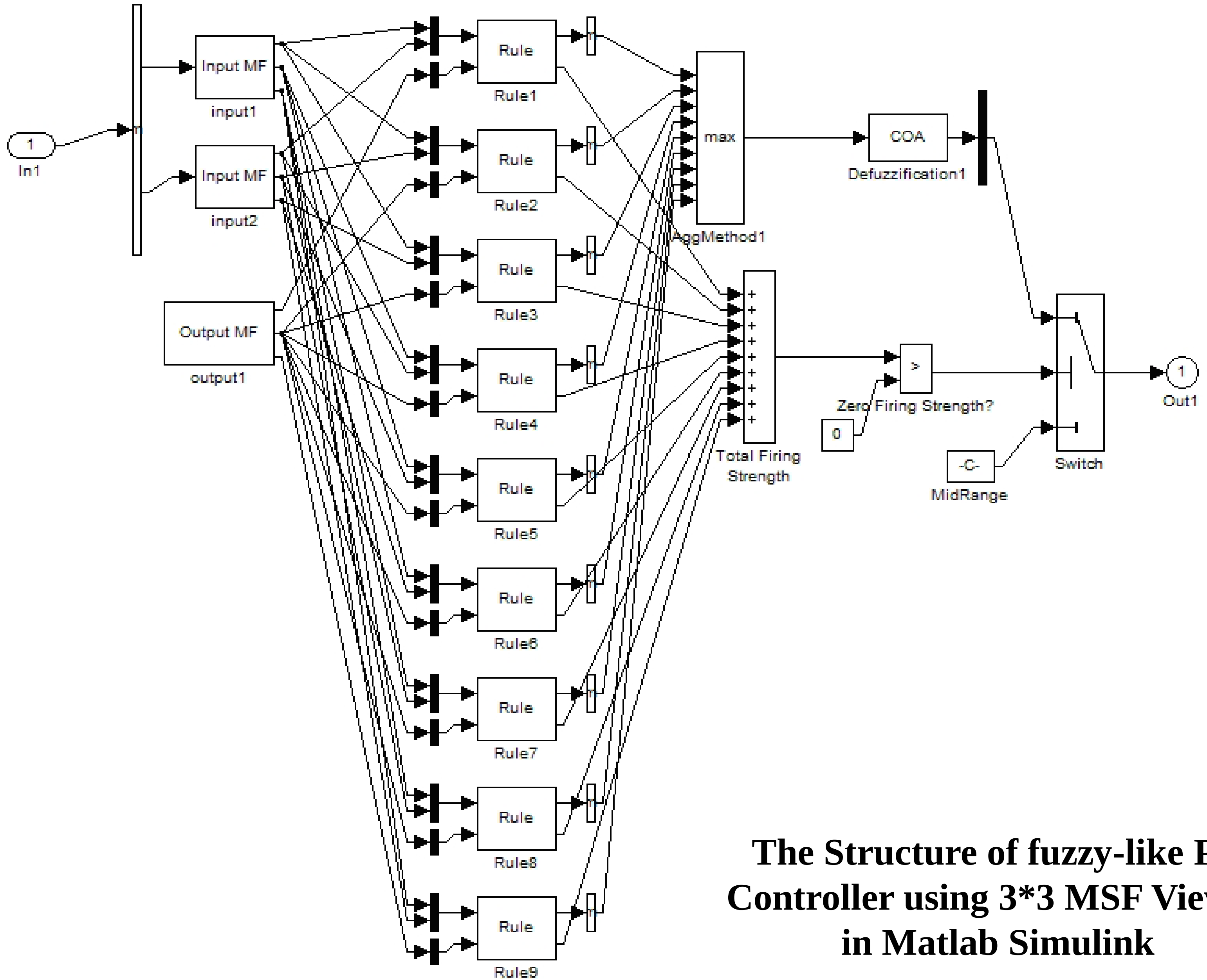

**(a)** **(b)**

**Version I of Fuzzy PD with Membership function of 3*3 (a) Inputs variables (b) Output Variable.**



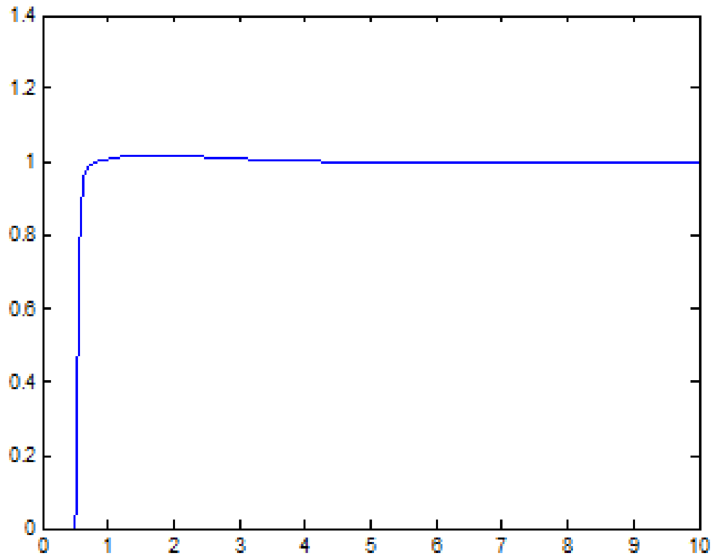

**(a)** **(b)**

**Version II of Fuzzy PD with Membership function of 3*3 (a) Inputs variables (b) Output Variable.**

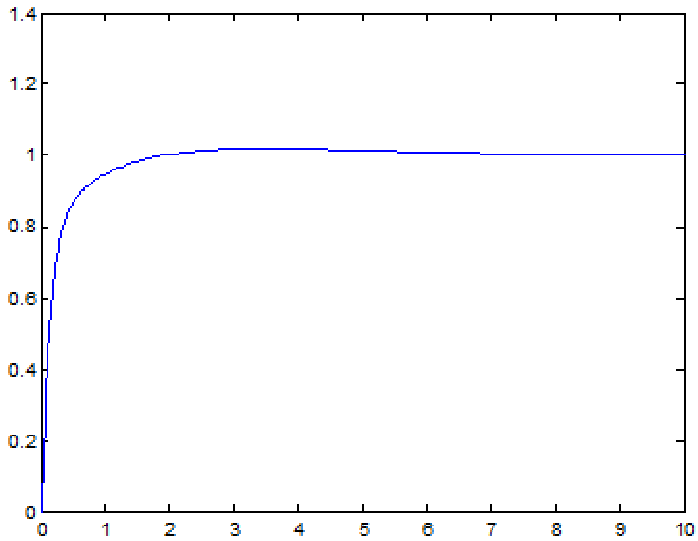



**The Structure of fuzzy-like PD Controller using 3*3 MSF Viewed in Matlab Simulink**

(a)




(b)



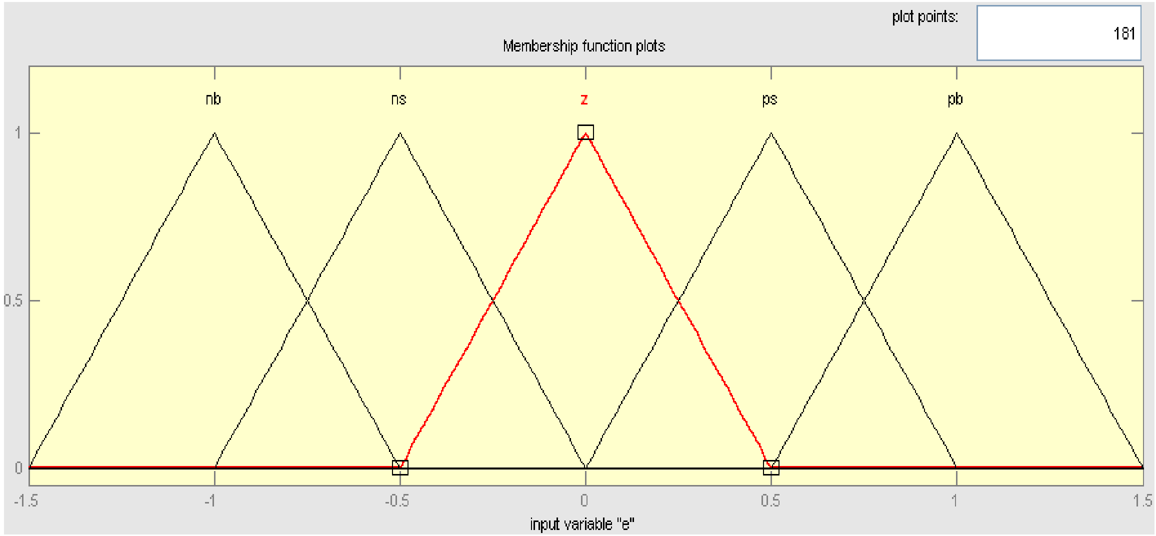

**The Response Obtained by Applying Fuzzy Logic-like PD Controller to the Position Control of an AC Motor for the 3*3 rule base (a) Version I and (b) Version II.**

## 2. Membership Function of 5*5 Rule Base

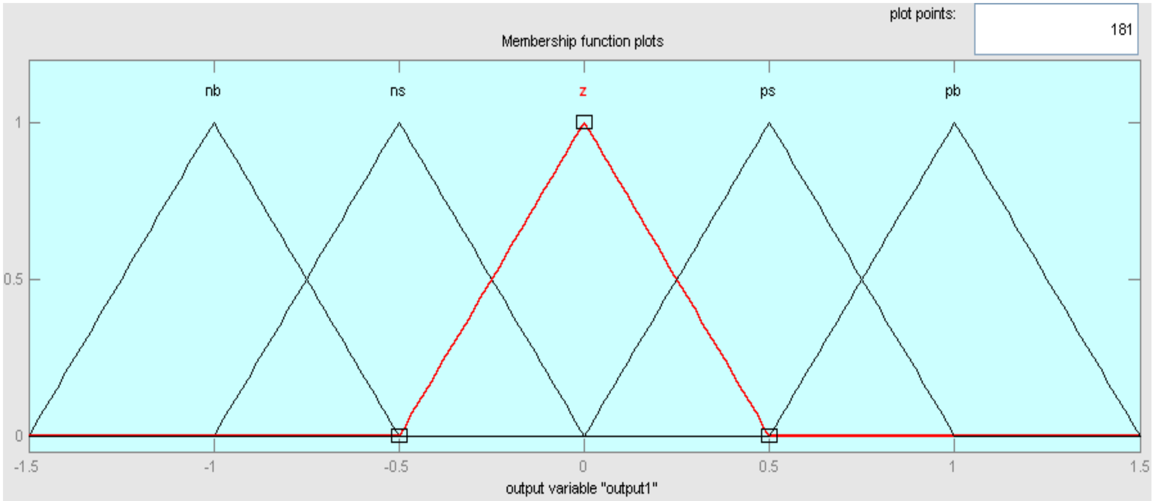
| | | e | | | | |
|---|---|---|---|---|---|---|
| | | NB | NS | Z | PS | PB |
| de | NB | PB | PB | PB | PS | Z |
| | NS | NS | PB | PS | Z | NS |
| | Z | PB | PS | Z | NS | NB |
| | PS | PS | Z | NS | NB | NB |
| | PB | Z | NS | NB | NB | NB |

**Fuzzy rule base of 25 rules for 5*5 two inputs fuzzy PD**

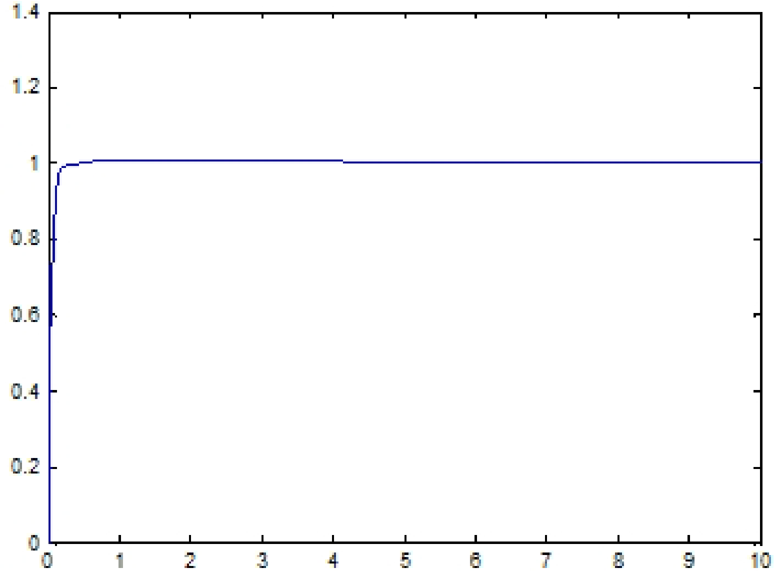
(a)




(b)




**Membership Function of 5*5 (a) Inputs Variables (b) Output Variable**

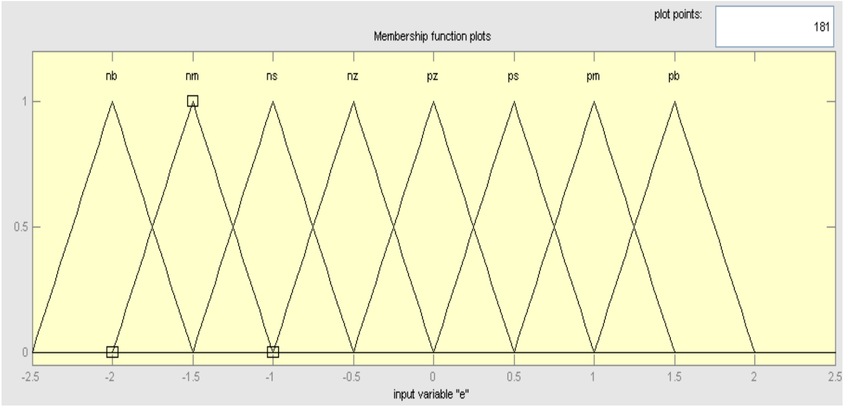



**The response by applying fuzzy logic –like PD controller to the position control of anAC motor for the 5*5 rule base.**

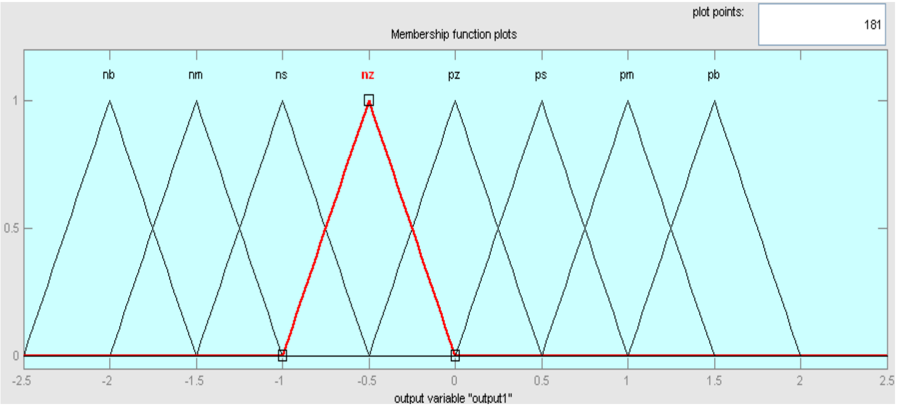
## 3. Membership Function of 8*8 Rule Base

| e/de | NB | NM | NS | NZ | PZ | PS | PM | PB |
|---|---|---|---|---|---|---|---|---|
| NB | NB | NB | NB | NM | NM | NS | NZ | PZ |
| NM | NB | NB | NM | NM | NS | NZ | PZ | PZ |
| NS | NB | NM | NM | NS | NZ | PZ | PZ | PS |
| NZ | NM | NM | NS | NZ | PZ | PZ | PS | PM |
| PZ | NM | NS | NZ | NZ | PZ | PS | PM | PM |
| PS | NS | NZ | NZ | PZ | PS | PM | PM | PB |
| PM | NZ | NZ | PZ | PS | PM | PM | PB | PB |
| PB | NZ | PZ | PS | PM | PM | PB | PB | PB |

**Fuzzy rule base of 64 rules for 8*8 two inputs fuzzy PD**



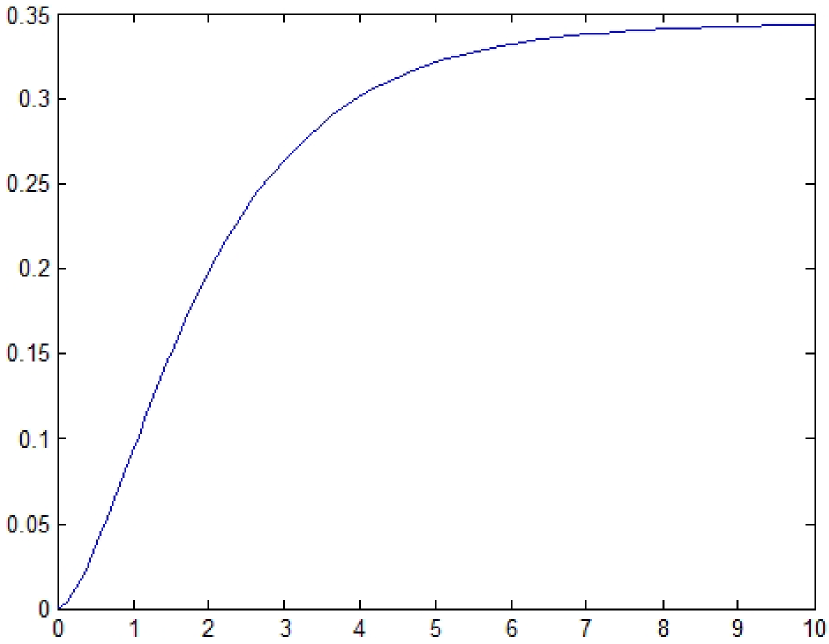

**(a)**

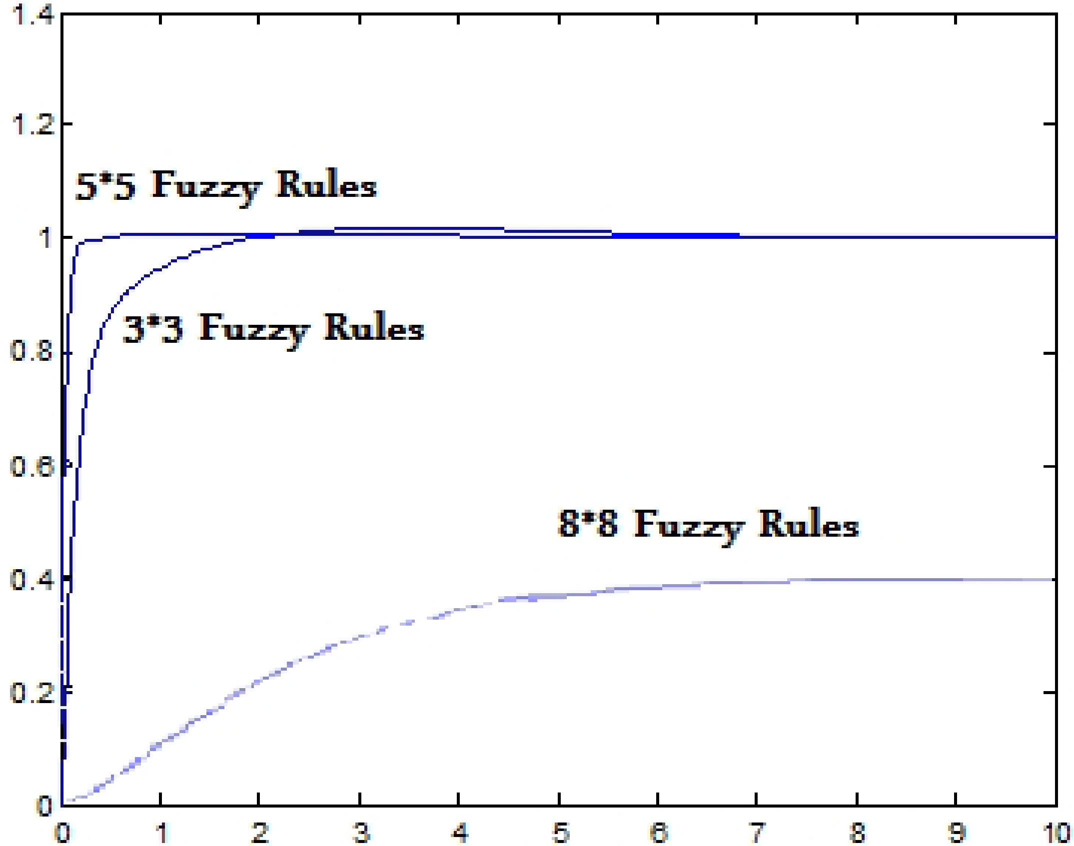



**(b)**

**The Membership Function of 8*8 (a) Inputs Variables (b) Output Variable**

**The response by applying fuzzy logic –like PD controller to the position control of an AC motor for the 8*8 rule base**

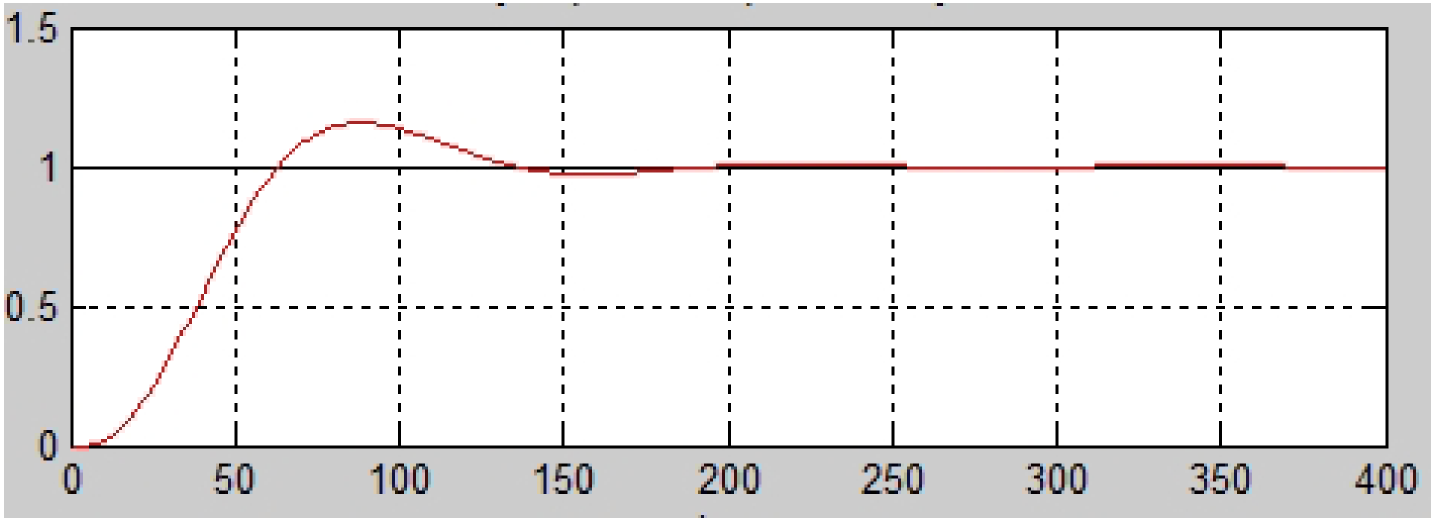




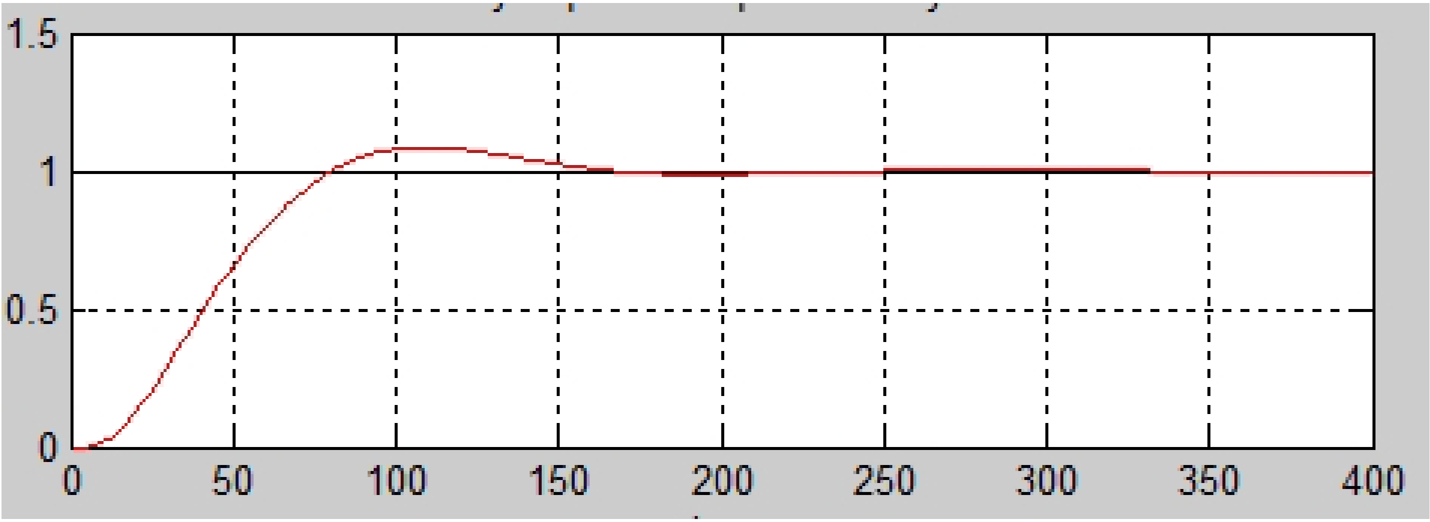

**The Response comparison by applying fuzzy logic –like PD controller to the position control of an AC motor for the 3*3, 5*5 and 8*8 rule base**

# Results of PI-PID Fuzzy Logic Controller in Matlab M-File

Due to limitation in Matlab Toolbox and Simulink for obtaining summation of error where used for designing PI fuzzy controller as well as the huge number of rule needed for designing PID fuzzy logic controller where in the case of 5 MSF that mean 5*5*5=125 rule needed. Therefore, M-file was used for designing PI_PID fuzzy controller where PID fuzzy controller designed using parallel structure of classical PID by using two inputs only with two PD fuzzy logic controller and by accumulating the output of the second PD fuzzy logic controller to form as PI fuzzy logic controller, where is obtained in discrete form x-axis are for the sampling rate (for real time should multiplied by 0.25 where is the value of the sampling rate). The simulation results of fuzzy like PID controller using 5 MSF and Rule base of 25 rules.




**System response by using fuzzy-like PI controller (for time values should multiply by 0.25)**

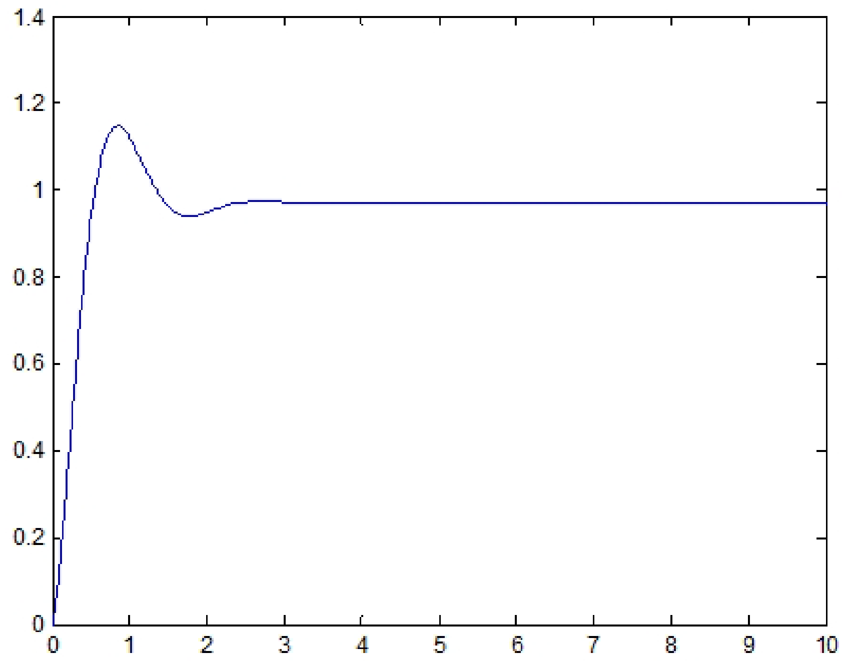



**System response by using fuzzy-like PID controller (for time values should multiply by 0.25)**

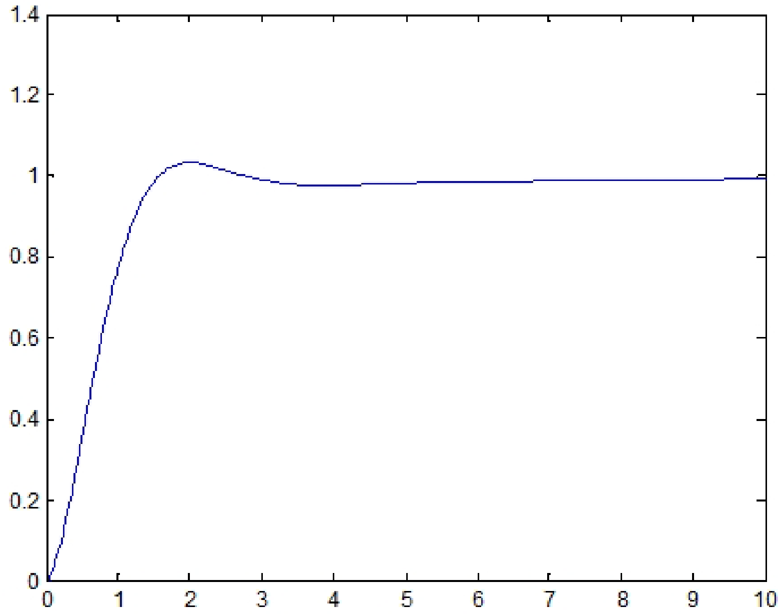
# PID Controller With Ziegler-Nichols(The First Method)

**Results of applying Ziegler-Nichols (The first method)**

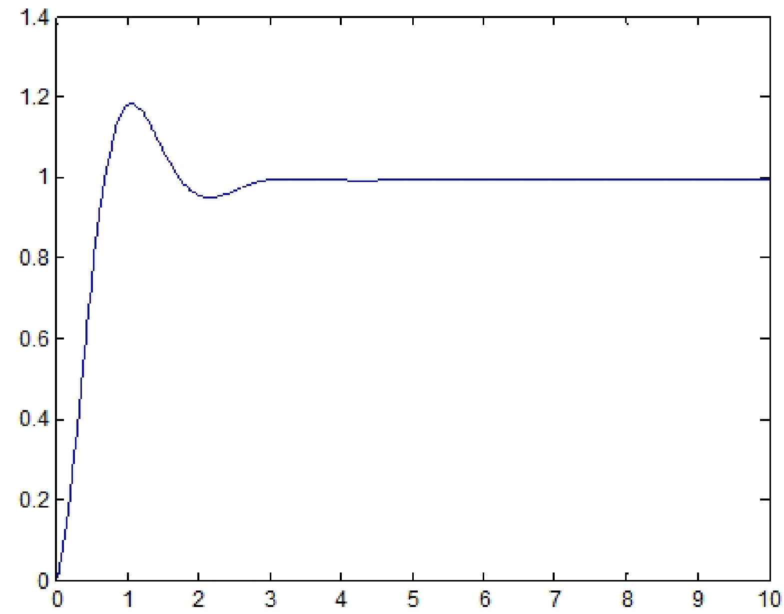
| kp | Ti | Td | ts | tr | Mp | e |
|---|---|---|---|---|---|---|
| 14.4 | 1 | 0.25 | 0.1 | 0.7 | 0.5 | 2.01 |
| 4.2 | 0.6 | 0.15 | 3.24 | 0.2 | 0.2 | 2.56 |
| 10.28 | 1.4 | 0.35 | 1.3 | 0.3 | 0.4 | 2.78 |




**(a) When KP=14.4, Ti=1 and Td=0.25**




**(b) When KP=4.2, Ti=0.6 and Td=0.15**




**(c) When KP=10.28, Ti=1.4 and Td=0.3**

**The result of applying on position control of an AC motor with different values of KP, Ti and Td**

The effective values of Kp is from 4.2 to 14.4, where the overshoot is increased from 0.2 to 0.5 and the steady state error is decreased from 2.5 to 2. The response performance of the system is improved with the fuzzy controllers' types. FPD eliminates the overshoot with small change on the error where the FPD with three Membership Function (3FPD) versions I is 0.005 and with 3FPD version II is zero overshoot, while the steady state error is zero in both versions of 3FPD. Fuzzy PD with 5 MSF (5FPD) has zero overshoot with steady state error equal to 0.005 and faster system response. Fuzzy PD with 8 MSF (8FPD) provides system response with zero overshoot versus big error bigger than 0.5 and slower system response due to the huge number of rules where was 64 fuzzy rules. Fuzzy PI with 5 MSF (5FPI) eliminates the error but with big overshoot equal to 0.2. FPID got the better response performance of the control system but slower than FPD.

# Conclusions

Higher execution speed versus is achieved by tuning the classical PID controller with the classical tuning methods. The fuzzy logic controller with fuzzy rules base of 5*5 provide the best control performance with respect to other types of controller Fuzzy controller provides good control performance versus faster execution speed. The design of the fuzzy control is based on human observation of a typical step response. Gradual increases in the controller gains, as the system response approaches zero error, provide improved system operation by increasing the speed of system and decrease the time delay with small rate of error.

# Thank you

# for listening