

# FUZZY LOGIC CONTROLLER (FLC) IMPLEMENTATION FOR SPACE VECTOR PULSE WIDTH MODULATED INDUCTION MOTOR DRIVE

<sup>1</sup>V.Raveendra Reddy, <sup>2</sup>Dr.V.C.Veera Reddy, <sup>3</sup>Dr. V.Chandra Jaganmohan

<sup>1</sup> Associate Professor, <sup>2</sup> Professor, <sup>3</sup> Professor

<sup>1</sup> Electrical And Electronics Engineering ,RSR Engineering College ,Kadanuthala,Kavali ,SPSR(Dst)

<sup>2</sup> Professor, Department of Electrical and Electronics Engineering, Annamacharya Institute of Tech & Science, Tirupati, Andhra Pradesh, India.

<sup>3</sup> Professor, Department of Electrical and Electronics Engineering, Institute of Aeronautical Engineering, Dundigal, Hyderabad-500043, Telengana state, India

**Abstract :** This paper presents Fuzzy Logic Controller(FLC) implementation for space vector pulse width modulated(SVPWM) fed Induction motor(I.M) drive using clusters.The MATLAB code for the clusters has been implemented.The proposed system has implemented and simulated using MATLAB/SIMULINK software.Time analysis (peak time, rise time, slew rate, settling time and overshoot ) have obtained and results are tabulated.

**IndexTerms** - Fuzzy Logic Controller(FLC), space vector pulse width modulated(SVPWM)

## I. INTRODUCTION

Soft computing techniques have been in existence for quite a large period of time with design of tools using modern technology, the need arises to put its usage to optimum effect. In this connection, a number of real world applications have been designed using these tools to a varying degree of success [6]. Fuzzy logic techniques attempt to imitate human thought processes in technical environments. The field of Fuzzy control has been making rapid progress in recent years. Fuzzy logic control (FLC) is one of the most successful applications of fuzzy set theory, introduced by L.A Zadeh in 1973 and applied in an attempt to control system that are structurally difficult to model. Since then, FLC has been an extremely active and fruitful research area with many industrial applications reported .In the last three decades, FLC has evolved as an alternative or complementary to the conventional control strategies in various engineering areas. Fuzzy control theory usually provides non-linear controllers that are capable of performing different complex non-linear control action, even for uncertain nonlinear Systems. Unlike conventional control, designing a FLC does not require precise knowledge of the system model such as the poles and zeroes of the system transfer functions. Imitating the way of human learning, the tracking error and the rate change of the error are two crucial inputs for the design of such a fuzzy control system .

## II. Fuzzy Logic Controller

Fuzzy systems are knowledge based or rule based systems. The heart of a fuzzy system is a knowledge base consisting of the so- called If-Then rules. A fuzzy If-Then statement in which some words are characterized by continuous membership functions. After defining the fuzzy sets and assigning their membership functions, rules must be written to describe the action to be taken for each combination of control variables. These rules will relate the input variables to the output variable using If-Then statements which allow decisions to be made. The structure of fuzzy controller is shown in Figure 2.1.

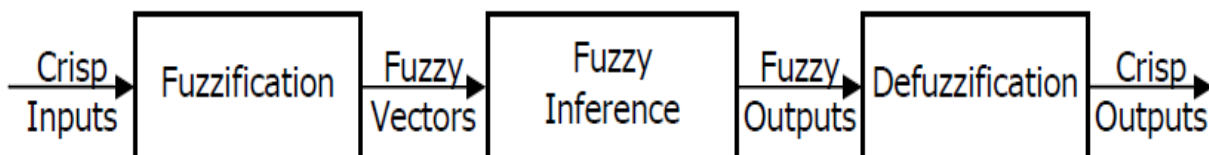


Fig 2.1 Fuzzy control structure

As seen in Figure 2.1, the fuzzy inference system consists of three main parts which can be illustrated as follows:

### 2.1 Fuzzification

Fuzzification means the process of converting an object into a fuzzy object. It is the process in which a crisp values are transformed into grades of membership for linguistic terms of fuzzy sets. A crisp set may be fuzzified by attaching a membership grade to each element of the set. The most commonly used method is the singleton fuzzifier[1].

### 2.1.1 Membership Functions

A membership function (MF) is a curve that defines how each point in the input space is mapped to a membership value (or degree of membership) between 0 and 1. Membership function is designated as  $\mu$ . The only condition a membership function must really satisfy is that it must vary between 0 and 1. The function itself can be an arbitrary curve whose shape we can define as a function that suits us from the point of view of simplicity, convenience, speed, and efficiency [6]

A classical set might be expressed as

- $A = \{x \mid x > 6\}$

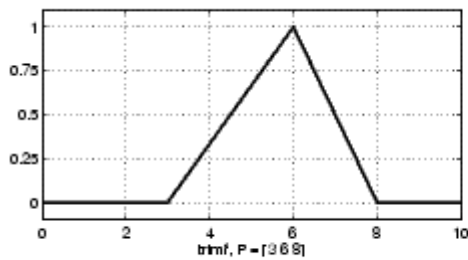
A fuzzy set is an extension of a classical set. If X is the universe of discourse and its elements are denoted by x, then a fuzzy set A in X is defined as a set of ordered pairs.

- $A = \{x, \mu_A(x) \mid x \in X\}$

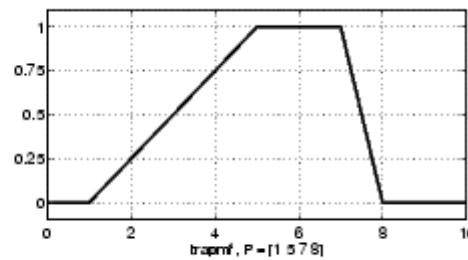
$\mu_A(x)$  is called the membership function (or MF) of x in A. The membership function maps each element of X to a membership value between 0 and 1.

The Fuzzy Logic Toolbox includes 11 built-in membership function types. These 11 functions are, in turn, built from several basic functions: piecewise linear functions, the Gaussian distribution function, the sigmoid curve, and quadratic and cubic polynomial curves.

- The simplest membership functions are formed using straight lines. Of these, the simplest is the *triangular* membership function, and it has the function name trimf. It's nothing more than a collection of three points forming a triangle.
- The *trapezoidal* membership function, trapmf, has a flat top and really is just a truncated triangle curve. These straight line membership functions have the advantage of simplicity.

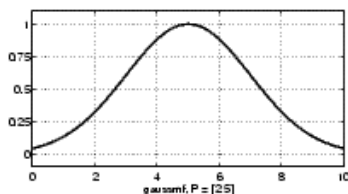


trimf

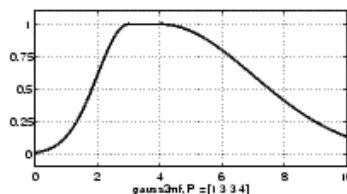


trapmf

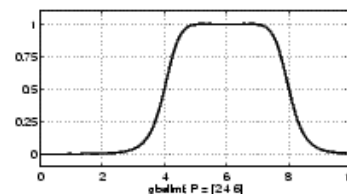
- Two membership functions are built on the *Gaussian* distribution curve: a simple Gaussian curve and a two-sided composite of two different Gaussian curves. The two functions are gaussmf and gauss2mf.
- The *generalized bell* membership function is specified by three parameters and has the function name gbellmf. The bell membership function has one more parameter than the Gaussian membership function, so it can approach a non-fuzzy set if the free parameter is tuned. Because of their smoothness and concise notation, Gaussian and bell membership functions are popular methods for specifying fuzzy sets. Both of these curves have the advantage of being smooth and nonzero at all points.



gaussmf

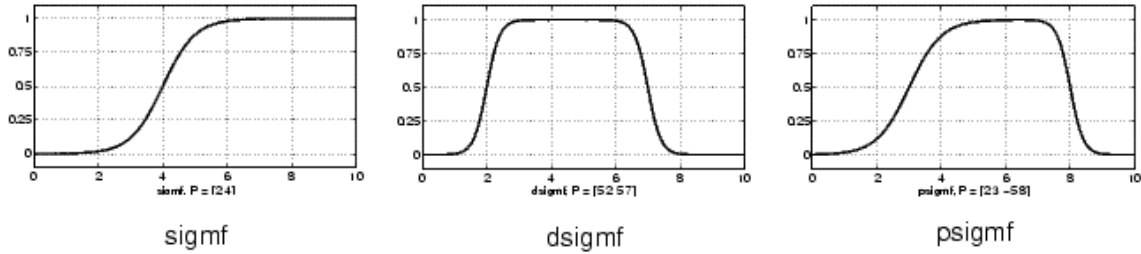


gauss2mf



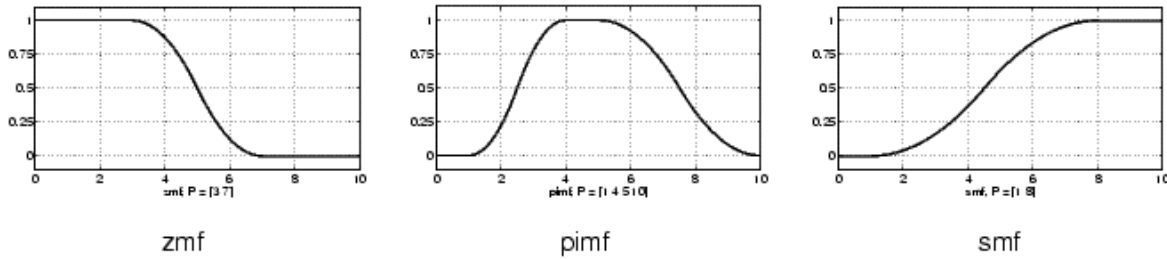
gbellmf

- Although the Gaussian membership functions and bell membership functions achieve smoothness, they are unable to specify asymmetric membership functions, which are important in certain applications. Next we define the *sigmoidal* membership function, which is either open left or right. Asymmetric and closed (i.e. not open to the left or right) membership functions can be synthesized using two sigmoidal functions, so in addition to the basic sigmf, we also have the difference between two sigmoidal functions, dsigmf, and the product of two sigmoidal functions psigmf.



Poly

nomial based curves account for several of the membership functions in the toolbox. Three related membership functions are the *Z*, *S*, and *Pi* curves, all named because of their shape. The function *zmf* is the asymmetrical polynomial curve open to the left, *smf* is the mirror-image function that opens to the right, and *pimf* is zero on both extremes with a rise in the middle.[6]



**2.2 The fuzzy inference engine**

Fuzzy inference engine is the process that relates input fuzzy sets to output fuzzy sets using if-then rules and fuzzy mechanism to derive a reasonable output form crisp or fuzzy inputs. There are two types of fuzzy inference systems: Mamdani-type and Sugeno-type [2].

Step 1: Computing the degree of fulfillment by:

$$\beta_i = \max[\mu_{A'}(x) \wedge \mu_{A_i}(x)], i=1,2,\dots,K \text{ -----(1)}$$

Step 2: Deriving the output fuzzy sets  $\mu_{B_i}'$  :

$$\mu_{B_i}'(y) = \beta_i \wedge \mu_{B_i}(y), y \in Y, i=1,2,\dots,K \text{ -----(2)}$$

Step 3: Aggregating the output fuzzy sets  $\mu_{B_i}'$  into a single fuzzy set  $B'$ :

$$\mu_{B'}(y) = \max_{1 \leq i \leq K} \mu_{B_i}'(y), y \in Y \text{ -----(3)}$$

**2.3 Defuzzification**

Defuzzification is the process from which a crisp value is produced from a fuzzy set through fuzzy inference. Although many defuzzification methods have been proposed, they are not resting on scientific bases; so defuzzification is considered an art rather than a science [1]. The following are the most famous defuzzification methods: centroid, average of maxima, midpoint of maxima, median, area, height, and maximal height . Among these methods, the most commonly used is the (centroid) defuzzification. This technique was developed by Sugeno in 1985, and it had very accurate results compared with other methods. The centroid defuzzification, which is shown in Figure 5.2, could be expressed as follows [3]:

$$X^* = \frac{\int \mu_i(x) x \, dx}{\int \mu_i(x) \, dx} \text{ ----- (4)}$$

where:  $x^*$  is the defuzzified output

$\mu_i(x)$  is the aggregated membership function

$x$  is the output variable

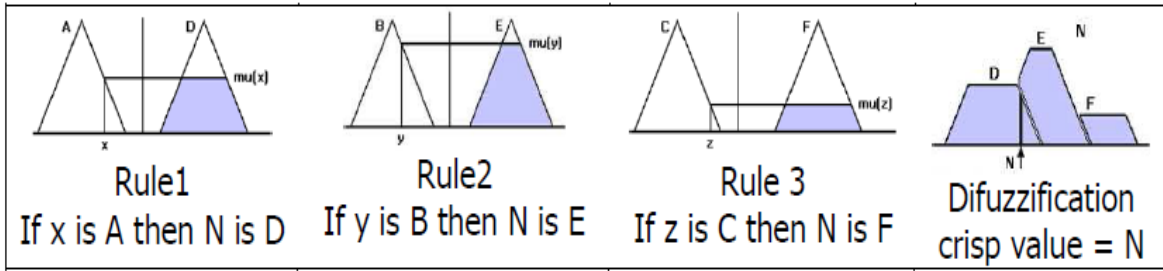


Fig 2.2: Centroid defuzzification using max-min inference

**2.3.1 Defuzzification Methods**

Fuzzy rule based systems evaluate linguistic if-then rules using fuzzification, inference and composition procedures. They produce fuzzy results which usually have to be converted into crisp output. To transform the fuzzy results in to crisp, defuzzification is performed. Defuzzification is the process of converting a fuzzified output into a single crisp value with respect to a fuzzy set. The defuzzified value in FLC (Fuzzy Logic Controller) represents the action to be taken in controlling the process.

The following are the known methods of defuzzification.

- Center of Sums Method (COS)
- Center of gravity (COG) / Centroid of Area (COA) Method
- Center of Area / Bisector of Area Method (BOA)
- Weighted Average Method
- Maxima Methods
  - First of Maxima Method (FOM)
  - Last of Maxima Method (LOM)
  - Mean of Maxima Method (MOM)

**2.3.1.1 Center of Sums (COS) Method**

This is the most commonly used defuzzification technique. In this method, the overlapping area is counted twice. The defuzzified value  $x^*$  is defined as

$$x^* = \frac{\sum_{i=1}^N x_i \cdot \sum_{k=1}^n \mu_{A_k}(x_i)}{\sum_{i=1}^N \sum_{k=1}^n \mu_{A_k}(x_i)}$$

Here, n is the number of fuzzy sets, N is the number of fuzzy variables,  $\mu_{A_k}(x_i)$  is the membership function for the k-th fuzzy set.

**2.3.1.2 Center of gravity (COG) / Centroid of Area (COA) Method**

This method provides a crisp value based on the center of gravity of the fuzzy set. The total area of the membership function distribution used to represent the combined control action is divided into a number of sub-areas. The area and the center of gravity or centroid of each sub-area is calculated and then the summation of all these sub-areas is taken to find the defuzzified value for a discrete fuzzy set.

For discrete membership function, the defuzzified value denoted as  $x^*$  using COG is defined as:

$$x^* = \frac{\sum_{i=1}^n x_i \cdot \mu(x_i)}{\sum_{i=1}^n \mu(x_i)}$$

Here  $x_i$  indicates the sample element,  $\mu(x_i)$  is the membership function, and n represents the number of elements in the sample.

**2.3.1.3 Center of Area / Bisector of Area Method (BOA)**

This method calculates the position under the curve where the areas on both sides are equal. The BOA generates the action that partitions the area into two regions with the same area.

$$\int_{\alpha}^{x^*} \mu_A(x) dx = \int_{x^*}^{\beta} \mu_A(x) dx, \text{ where } \alpha = \min \{x | x \in X\} \text{ and } \beta = \max \{x | x \in X\}$$

### 2.3.1.4 Weighted Average Method

This method is valid for fuzzy sets with symmetrical output membership functions and produces results very close to the COA method. This method is less computationally intensive. Each membership function is weighted by its maximum membership value. The defuzzified value is defined as :

$$x^* = \frac{\sum \mu(x).x}{\sum \mu(x)}$$

Here  $\sum$  denotes the algebraic summation and  $x$  is the element with maximum membership function.

## 2.3 FUZZY IMPLEMENTATION:

The implementation of FLC could be summarized in the following steps

1. Determination which of system dynamic parameters will be used as input, and which will be used as output of the fuzzy controller. For example: rotor speed and rotor acceleration are the inputs, while voltage and frequency are the outputs.
2. Choosing proper linguistic fuzzy variables to describe the membership functions for input and output variables of the fuzzy controller. These variables are used to transform numerical values into fuzzy quantities. For example: (NL) is used for (negative large), and (PM) is used for (positive medium).
3. Building the fuzzy control rules that relate input variables to output variables using (if-then) statements.
4. Finally, a fuzzy centroid is used to calculate the fuzzy control output.

## 2.4 TYPES OF FIS SYSTEMS:

1. MAMDANI-TYPE FIS
2. SUGENO-TYPE FIS

### 2.4.1 MAMDANI-TYPE FIS VS. SUGENO-TYPE FIS

Mamdani method is widely accepted for capturing expert knowledge. It allows us to describe the expertise in more intuitive, more human-like manner. However, Mamdani-type FIS entails a substantial computational burden. On the other hand, Sugeno method is computationally efficient and works well with optimization and adaptive techniques, which makes it very attractive in control problems, particularly for dynamic non linear systems. These adaptive techniques can be used to customize the membership functions so that fuzzy system best models the data.

The most fundamental difference between Mamdani-type FIS and Sugeno-type FIS is the way the crisp output is generated from the fuzzy inputs. While Mamdani-type FIS uses the technique of defuzzification of a fuzzy output, Sugeno-type FIS uses weighted average to compute the crisp output. The expressive power and interpretability of Mamdani output is lost in the Sugeno FIS since the consequents of the rules are not fuzzy . But Sugeno has better processing time since the weighted average replace the time consuming defuzzification process. In this unit sugeno method is used to implement the speed control of induction motor.

### 2.4.2 TAKAGI-SUGENO CONTROL SCHEME

Takagi and Sugeno [6] proposed a new type of fuzzy model (TS model) which has been widely used in many disciplines, especially in the control of dynamical systems, such as induction motors, DC motors, AC motors, etc. The main feature of a Takagi-Sugeno fuzzy model is to express the local dynamics of each fuzzy implication (rule) by a linear system model. The overall fuzzy model of the system is achieved by fuzzy “blending” of the linear system models. These TS models use fuzzy rules with fuzzy antecedents and functional consequent parts, thereby qualifying them as mixed fuzzy or nonfuzzy models [5].

## 2.5 CLUSTERING METHODS:

**Cluster analysis** or **clustering** is the task of grouping a set of objects in such a way that objects in the same group (called a **cluster**) are more similar (in some sense) to each other than to those in other groups (clusters).

For clustering, many algorithms are adopted in literature [5], The three fuzzy clustering algorithms are subtractive, C-means and K-means clustering algorithms. The subtractive algorithm is used to determine the cluster number, whereas the C-means and the K-means will be exploited to generate the cluster centers then to construct the clusters. Here we use C-means clustering algorithm.

### 2.5.1 C-means clustering

The fuzzy c-means (FCM) algorithm is a clustering algorithm developed by Dunn, and later on improved by Bezdek[6]. It is useful when the required number of clusters are pre-determined; thus, the algorithm tries to put each of the data points to one of the clusters. What makes FCM different is that it does not decide the absolute membership of a data point to a given cluster; instead, it calculates the likelihood (the degree of membership) that a data point will belong to that cluster. Hence, depending on

the accuracy of the clustering that is required in practice, appropriate tolerance measures can be put in place. Since the absolute membership is not calculated, FCM can be extremely fast because the number of iterations required to achieve a specific clustering exercise corresponds to the required accuracy. It consists in producing an optimal partition by minimizing the objective function  $J$  on the basis of the following process.

- 1) Initialize arbitrarily the Fuzzy membership matrix  $\mu_{ik}$
- 2) Calculate the cost function  $J$  by (1).

$$J = \sum_{i=1}^c J_i = \sum_{i=1}^c \sum_{k=1}^N (\mu_{ik}^{\xi}) d_{ik}^2 \quad (1)$$

Here,  $N$  is the number of data points,  $C$  is the number of clusters required.

- 3) Estimate the clusters centers  $c_i$  by the Equation (2).

$$c_i = \frac{\sum_{k=1}^N (\mu_{ik}^{\xi}) x_k}{\sum_{k=1}^N (\mu_{ik}^{\xi})} \quad (2)$$

- 4) Updates the membership functions as the relation (3).

$$\mu_{ik} = \frac{1}{\sum_{j=1}^c \left( \frac{d_{ik}}{d_{jk}} \right)^{2/(\xi-1)}} \quad (3)$$

- 5) Recalculate the cost function  $J$ . If  $J$  is less than a threshold, the process will be ended. If not, return to step3.

## 2.6 Design of the Fuzzy Logic Controller using MATLAB

While simulating the block diagram in MATLAB/SIMULINK®, the Fuzzy Logic Controller has to be programmed according to the aforementioned rules and knowledge base. The program is saved as an FIS file and it is later embedded into the Fuzzy Logic Controller[6]. This FIS program can be checked with the help of FIS editor in MATLAB itself. The steps for the following are shown below, along with the membership functions[1], the rules and the surface plot viewed with the help of the FIS editor.

### Step 1:

1. The program for designing of the Fuzzy Logic Controller is written in a word file.
2. The definitions for all the shown membership functions are written in the program.
3. The rules are written in the program according to the syntax provided by MATLAB.
4. The document is saved with the extension **.fis**.

The If- Then Rules of the Rule Base used for the design of the Fuzzy Logic Controller are as follows:

**IF (in 1 is in 1 cluster1 ) AND (in 2 is in 2cluster1) THEN (out1 is out1cluster 1)(out 2 is out 2 cluster1)**

The program for designing the Fuzzy Logic Controller using the FIS editor in MATLAB/SIMULINK® is as follows

```
[System]
Name='control'
Type='sugeno'
Version=2.0
NumInputs=2
NumOutputs=2
NumRules=1
AndMethod='prod'
OrMethod='probor'
ImpMethod='prod'
AggMethod='max'
```

DefuzzMethod='wtaver'

```
[Input1]
Name='in1'
Range=[-0.368095166534971 2.00919477320698]
NumMFs=1
MF1='in1cluster1':'gaussmf',[0.168099783723809 0.0055357952024303]
```

```
[Input2]
Name='in2'
Range=[-1.64940152085574 0.99999998875]
NumMFs=1
MF1='in2cluster1':'gaussmf',[0.187340977352809 -0.014542452479744]
```

```
[Output1]
Name='out1'
Range=[-0.528545879469537 2.00919477320698]
NumMFs=1
MF1='out1cluster1':'linear',[0.578338023030855 -0.13669671973754 0.000419627836359093]
```

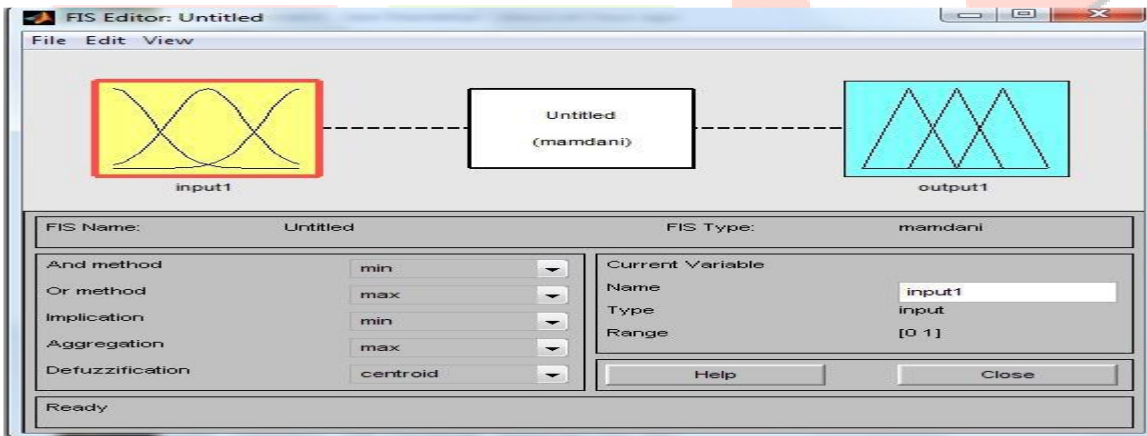
```
[Output2]
Name='out2'
Range=[-1.504078666473 0.654186380001725]
NumMFs=1
MF1='out2cluster1':'linear',[0.0874451302902101 0.456020400891536 0.00380660461800737]
```

```
[Rules]
1 1, 1 1 (1) : 1
```

**Step 2:**

The .fis file is now to be loaded in the FIS editor to view the membership functions, the rules and the rule surface plot.

1 On the command window of MATLAB fuzzy is typed to open the FIS editor



➤ Click **File > Import > From file...** and then browse the .fis file to open **FIS editor**:

**Rules**

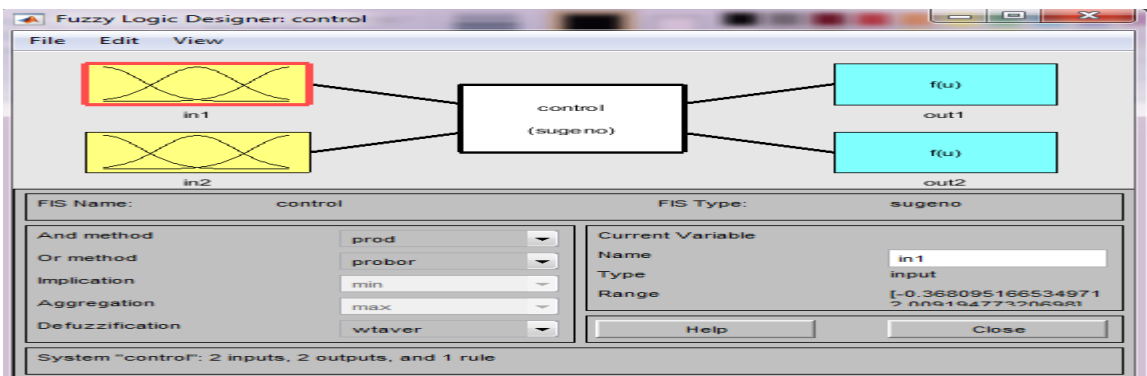


Fig 2.3 : FIS editor window in MATLAB

- Click on any of the input or output to view the respective membership functions. The membership functions for inputs 1 and 2 and for output 1 and output 2 are shown in Fig. In the **FIS editor: rules** window click on **View > Surface** to view the three dimensional plot of the control surface. This plot is shown in Fig 2.8.
- Then in the **FIS editor: rules** window click on **View > Rules** to see the rules. The inputs can be changed in the window and respective outputs can be viewed as shown in fig 2.9.

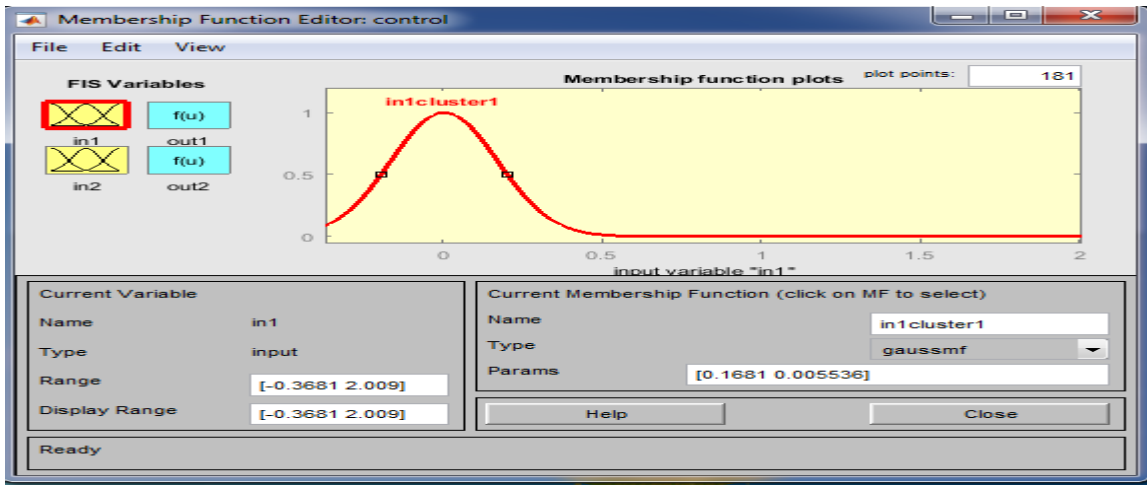


Fig 2.4 : Membership function for the input 1

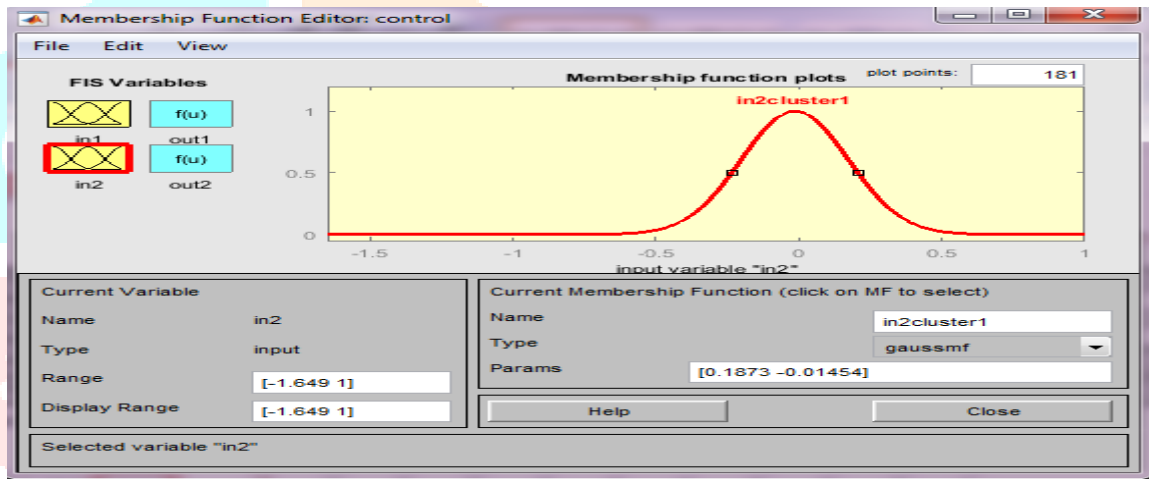


Fig 2.5: Membership function for the input 2

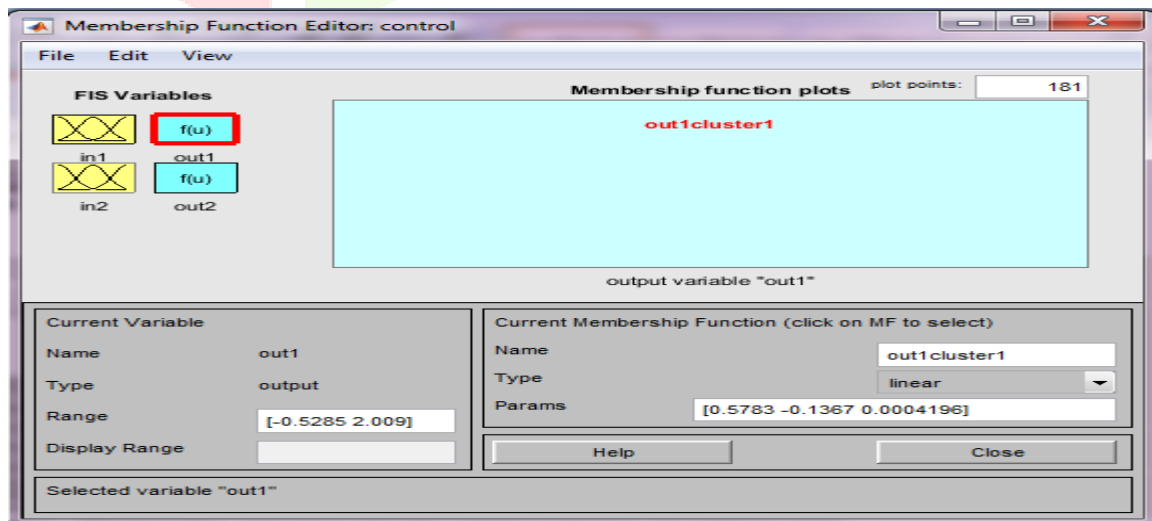


Fig 2.6 : Membership function for the output 1



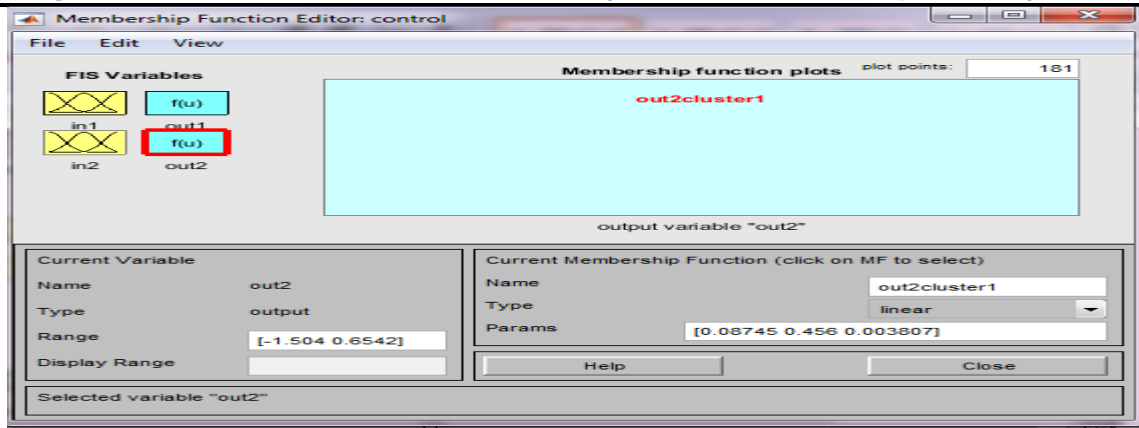


Fig 2.7 : Membership function for the output 2

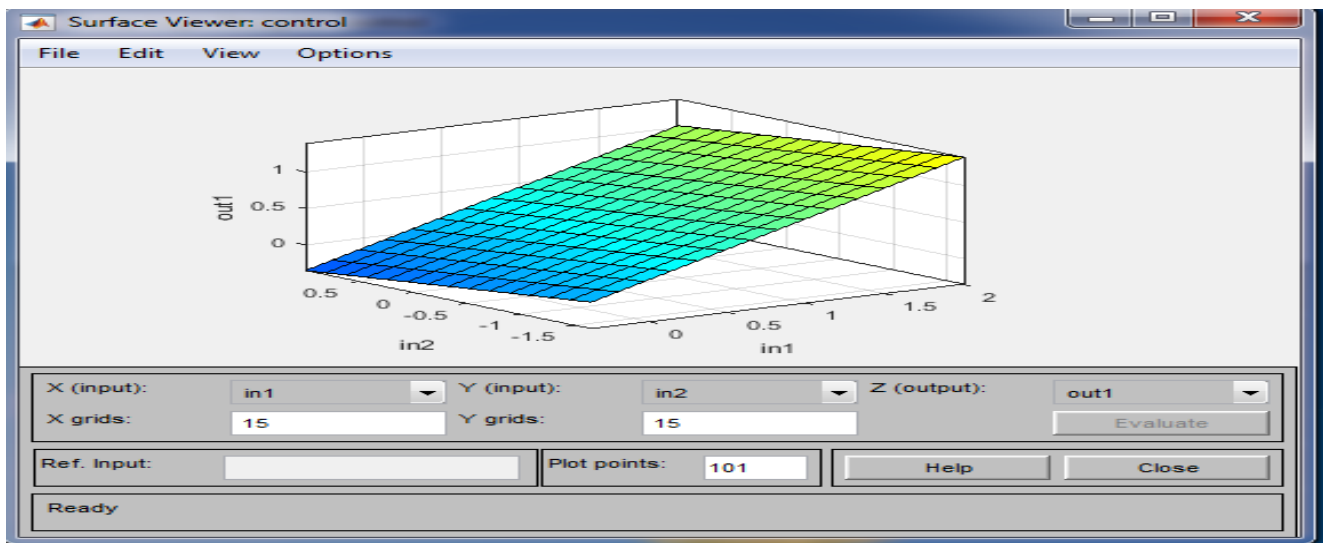


Fig 2.8 : Three dimensional plot of the control surface

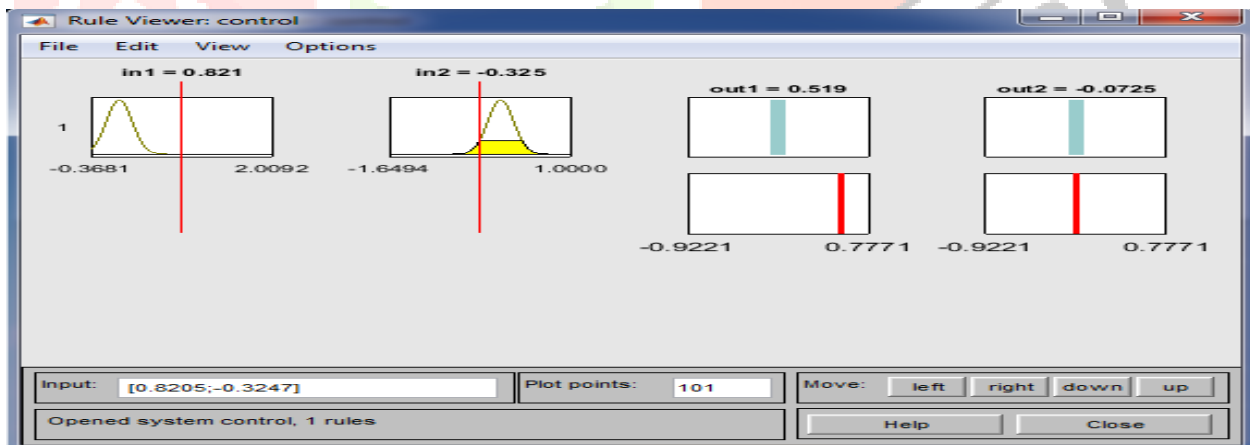


Fig 2.9: Rule viewer

### 2.7 Advantages of Fuzzy Logic Controller

The advantages provided by a FLC are listed below:

- It is simple to design.
- It provides a hint of human intelligence to the controller.
- It is cost effective.
- No mathematical modeling of the system is required.
- Linguistic variables are used instead of numerical ones.
- Non-linearity of the system can be handled easily.
- System response is fast.
- Reliability of the system is increased.
- High degree of precision is achieved.

### III. Block diagram and simulink model

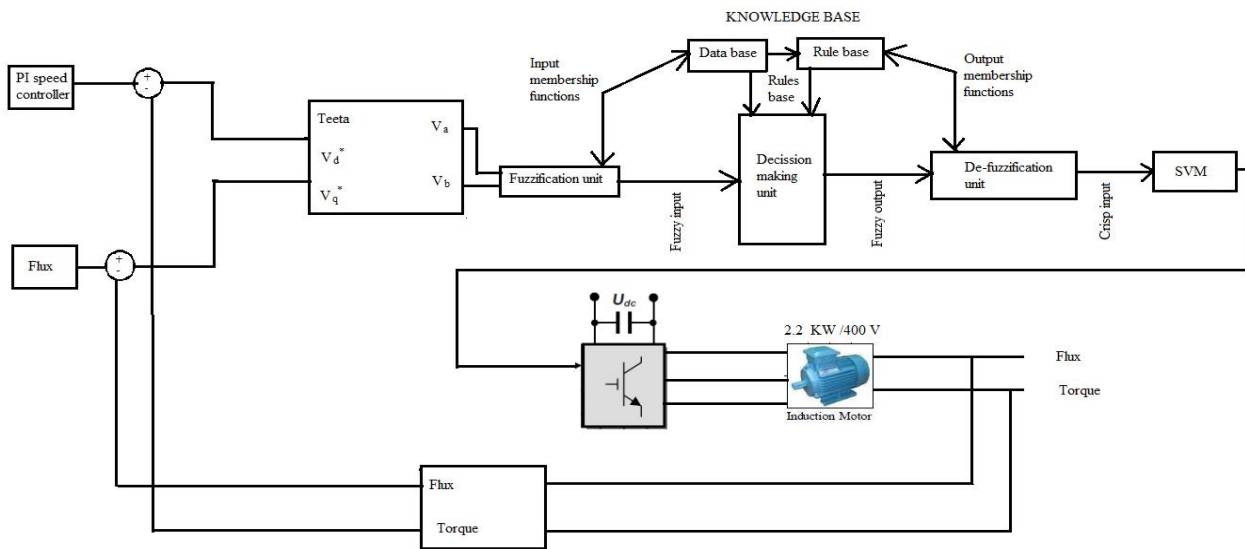


Fig 2.10: Block diagram of an induction motor using fuzzy logic controller.

### IV. MATLAB SIMULINK DIAGRAM:

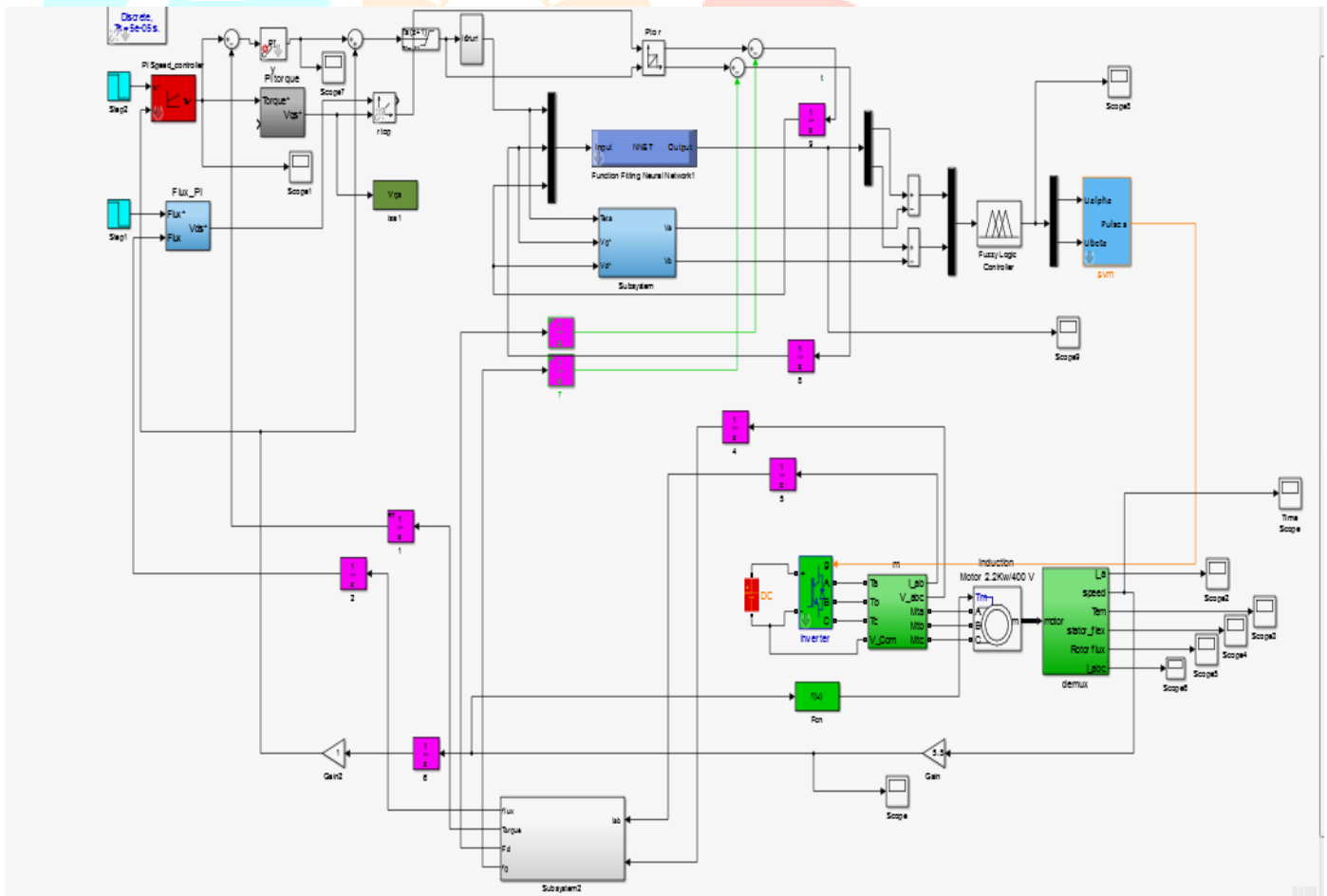


Fig 2.11 : Block diagram of an induction motor using fuzzy logic controller.

The block diagram in Fig 2.11 was simulated and the plots for speed, current, and torque, stator flux, stator current, rotor flux using Fuzzy Logic Controller were observed.

**V Results and Conclusion:**

Speed control of induction motor using Fuzzy Logic Controller controller has been done by simulating matlab/simulink models. Matlab code for the phasors Va (cluster 1) and Vb(cluster 2) has been written and the results of time analysis (peak time, rise time, slew rate, settling time and overshoot) are presented in table 1.

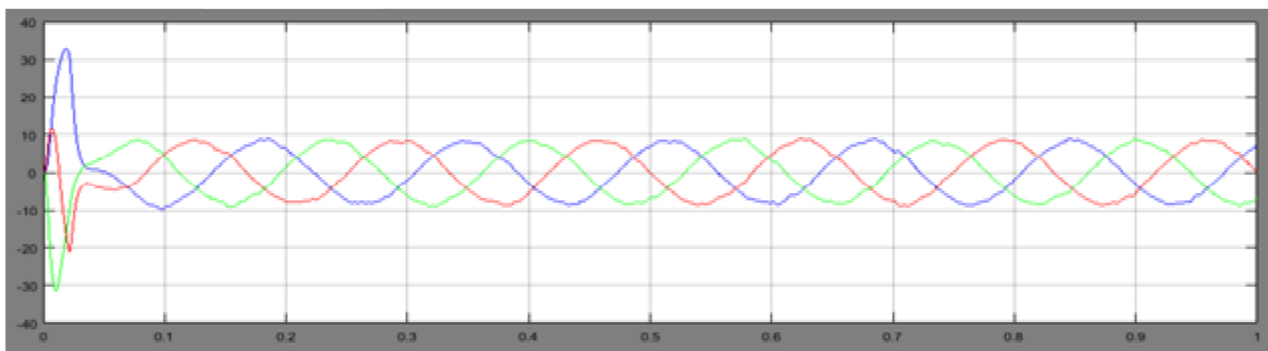


Figure 2.12 :Steady state plot of Stator current for svpwm fed NNDTC I.M drive

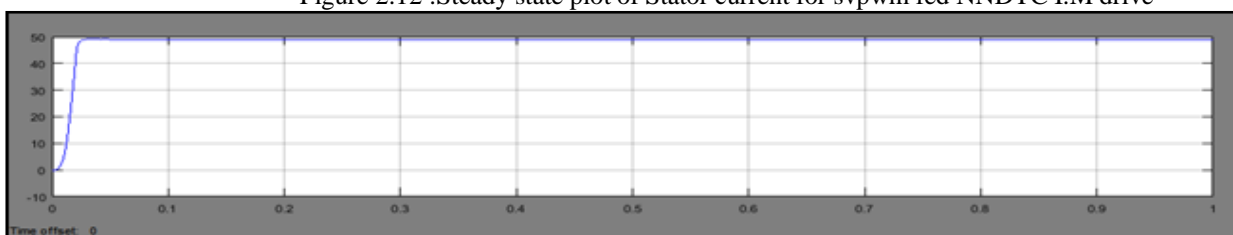


Figure 2.13: Steady state plot of rotor speed for svpwm fed FLC I.M drive

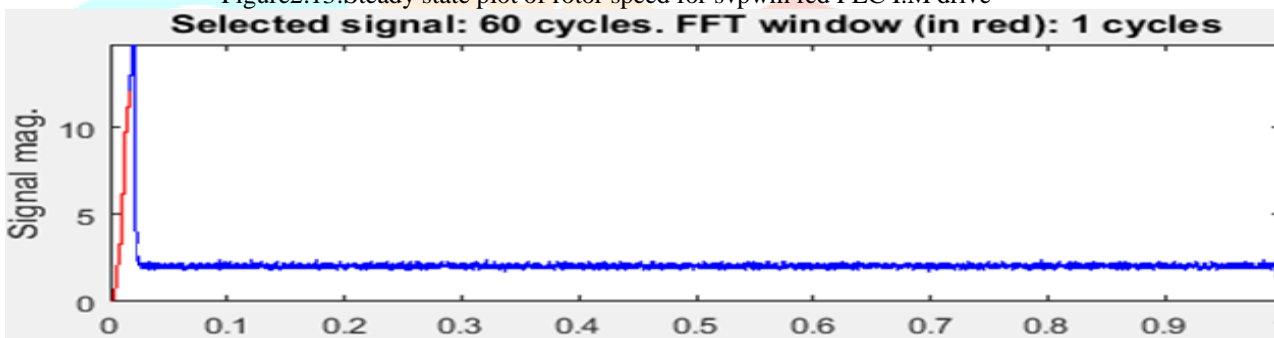


Figure 2.14 :Steady state plot of torque waveform of svpwm fed FLC I.M drive

**Table 1:** peak time, rise time, slew rate, settling time and overshoot of SVPWM fed FLC I.M drive.

	Speed (rpm)	tp (sec)	tr (sec)	Slew rate(v/sec)	ts(sec)	Over shoot %
FLC	1500	0.00877	0.005963	896.002	0.008663	_____

**REFERENCES:**

[1] Fuzzy Logic Based Speed Control System for Three- Phase Induction Motor, Marwan A. Badran, Mostafa A. Hamood, Waleed F. Faris, analele universităţii “eftimie murgu” resişa anul xx, nr. 1, 2013, issn 1453 – 7397

[2] Comparison of Mamdani-Type and Sugeno-Type Fuzzy Inference Systems for Air Conditioning System, Arshdeep Kaur, Amrit Kaur, International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-2, Issue-2, May 2012

[3] A method for defuzzification based on centroid point, Rahim Saneifard, Rasoul Saneifard, TJFS: Turkish Journal of Fuzzy Systems (eISSN: 1309–1190) An Official Journal of Turkish Fuzzy Systems Association Vol.2, No.1, pp. 36-44, 2011

[4] Fuzzy Model Identification Using Kernel-Density-Based Clustering Szymon Łukasik Piotr Andrzej Kowalski Małgorzata Charytanowicz Piotr Kulczycki. Systems Research Institute, Polish Academy of Sciences ul. Newelska 6, PL-01-447 Warsaw, Poland

[5] Fan, J.L. and Li, J. (2014) A Fixed Suppressed Rate Selection Method for Suppressed Fuzzy C-Means Clustering Algorithm.

[6] Abid A., Ben Mabrouk Z., Ben Hamed M., Sbita L., (2013) Multiple Lunberger Observer for an Induction Motor Represented by Decoupled Multiple Model. 10th International Multi-Conference on Systems, Signals & Devices (SSD)