CHAPTER 5

DESIGN OF FLC IN COMBINATION WITH HPWM AND SVPWM FOR FOC OF PMSM

5.1 INTRODUCTION

PMSM drives are widely used in applications where smooth torque is required. However, the main disadvantage of PMSM is torque ripple that leads to mechanical vibration and acoustic noise. For applications that require precise tracking, the machine should be free of torque ripples. The advanced control technique like vector control approach provides better dynamic response than the conventional scalar control to obtain lesser pulsations in torque at constant switching frequency. The system performance is judged from the outer loop in the field oriented control technique. The conventional PI controllers provide constant and fixed gain values in the proportional and integral co-efficient, and therefore it is usually preferred in the automation control systems. The characteristics of these controllers are affected by the disturbance in the load and speed and parameter variations. Such issues could be resolved by using human like FLC algorithm which are based on rule base analysis.

In this chapter two new instantaneous torque control schemes, namely, FLC with HPWM and FLC with SVPWM respectively has been put forth to reduce torque ripples in PMSM driven by FOC. The design, analysis and simulation of the proposed systems are done using SIMULINK under MATLAB and the simulation results discussed.
5.2 PROPOSED SYSTEM

FOC also known as vector control or decoupling control aims to control the motor flux and the torque to track the command trajectory of the machine and load variations. It is based on three major points,

- Machine current and voltage space vectors.
- Transformation of a three phase speed and time dependent system into a two co-ordinate time invariant system.
- Effective Pulse Width Modulation pattern generation.

FOC transformation requires two constants, namely, torque component (q coordinate) and the flux component (aligned with d coordinate). As FOC is simply based on projections, the control structure handles instantaneous electrical quantities. This makes the control accurate in every working operation (steady state and transient) and independent of the limited bandwidth mathematical model. The FOC thus solves the classic scheme problems, in the following ways, namely,

1. The ease of reaching constant reference (torque component and flux component of the stator current)

2. The ease of applying direct torque control, because in the (d,q) reference frame the expression of the torque is,

\[ T \propto \lambda_{dm} i_{qs} \]  \hspace{1cm} (5.1)

By maintaining the amplitude of the rotor flux (\( \lambda_{dm} \)) at a fixed value, a linear relationship between the torque and torque component (\( i_{qs} \)) is
established. The torque can then be controlled by controlling the torque component of the stator current vector.

The block diagram of the proposed FLC with HPWM based FOC of PMSM is shown in the Figure 5.1. The block diagram of proposed FLC with SVPWM based FOC of PMSM is shown in the Figure 5.2. Single phase AC supply is given to the inverter through the rectifier. The Inverter feeds the PMSM which drives the load. The three phase stator currents $i_a, i_b, i_c$ are measured.

The Clarke transformation is applied to determine the stator current projection in the two co-ordinates a,b stationary frame. The transformation from the three phase system into the $(\alpha,\beta)$ two dimension orthogonal system is given by the relations. The park transformation is then applied to obtain this projection in the d, q-rotating frame. The torque is sensed from the motor and compared with the reference torque. The error signal is given to the FLC.

The inputs to the FLC are Torque error ($e$) and change in torque error ($\delta\omega$). The FLC converts the crisp error input and change in torque error into fuzzy variables which are understandable to the fuzzy logic system and then are mapped into linguistic labels to obtain the output.

The FLC calculates a reference torque, which is proportional to the quadrature-axis stator current component $i_{sqref}$.

- The proposed FLC with HPWM and FLC with SVPWM based FOC of PMSM, which is shown in Figure 5.1 and Figure 5.2 respectively, require only two PI controllers. One for $i_d$ current control and one for $i_q$ Current control and one FLC torque control.
- The PI controllers, Park and Clarke transformations, Inverse Park and Inverse Clarke transformations, HPWM and SVPWM are already discussed in chapter 3, is used in this chapter also.

![Block Diagram of Proposed FLC with HPWM Based FOC of PMSM](image)

**Figure 5.1** Block Diagram of Proposed FLC with HPWM Based FOC of PMSM
Figure 5.2  Block Diagram of Proposed FLC with SVPWM Based FOC of PMSM
5.2.1 FLC

5.2.1.1 Introduction

Fuzzy logic (FL) is defined as multi-valued logic which deals with problems that have fuzziness or vagueness. FL is a problem solving control system methodology that lends itself to implementation in systems ranging from simple, small, embedded micro controllers to large, networked, multi channel PC or workstation-based data acquisition and control systems. It can be implemented in hardware, software, or a combination of both. FL provides a simple way to arrive at a definite conclusion based upon vague, ambiguous, imprecise, noisy, or missing input information. FL approach to control problems mimic the way a person would make decisions. However, it is faster. The control algorithm of a process that is based on FL is defined as a fuzzy control. The basic concept behind FLC is to utilize the expert knowledge and experience of a human operator for designing a controller an application process whose input-output relationship is given by a collection of fuzzy control rules using linguistic variables instead of a complicated dynamic model (Mendel 1995).

Advantages of FL include,

i. Conceptually easy to understand.

ii. Flexible.

iii. It can be merged with the conventional control system techniques.

iv. FL system is based on natural language.
5.2.1.2 Fuzzy Inference System (FIS)

FIS are also known as fuzzy rule-based systems, fuzzy model, fuzzy expert system, and fuzzy associative memory. This is a major unit of a fuzzy logic system. The block diagram of FIS is shown in the Figure 5.3. A FIS basically consists of a formulation of the mapping from a given input set to an output set using FL. The decision-making is an important part in the entire system. The FIS formulates suitable rules and, the decision is made based upon the rules. This is mainly based on the concepts of the fuzzy set theory, fuzzy IF–THEN rules, and fuzzy reasoning. FIS uses “IF... THEN...” statements, and the connectors present in the rule statement are “OR” or “AND” to make the necessary decision rules. The basic FIS can take either fuzzy inputs or crisp inputs, but the outputs are almost always fuzzy sets. Fuzzy inference system consists of a fuzzification interface, a rule base, a database, a decision-making unit, and finally a defuzzification interface (Mendel 1995).

A fuzzy inference process consists of the following five steps, namely,

1. Fuzzification of input variables.
2. Application of Fuzzy operator in the antecedent of the rule.
3. Implication from the antecedent to the consequent.
4. Aggregation of the consequents across the rules.
5. Defuzzification of output variables.
There are two important types of FIS. They are,

1. Mamdani FIS
2. Sugeno FIS

The difference between the two methods lies in the consequent of fuzzy rules. Fuzzy sets are used as rule consequents in Mamdani FIS and linear functions of input variables are used as rule consequents in Sugeno’s method.

The five graphical tools for building, editing and observing FIS in the Fuzzy Logic Toolbox in MATLAB software is shown in Figure 5.4.

i. The FIS editor helps to display the general information about fuzzy inference systems.

ii. The Membership Function (MF) editor displays and permits editing of all the MFs associated with the input and output variables.
iii. Once the rule matrix is designed and the fuzzy variables are defined in the FIS editor, the actual rules can be constructed by the Rule editor.

5.2.1.3 Membership Functions

Fuzziness in a fuzzy set is characterized by its Membership Function (MF). It classifies the element in the set, whether it is discrete or continuous. Each point in the input axis is mapped with the output space with the help of membership values which lie between 0 and 1 and is represented as a curve of Membership function.

A MF can have different shapes. The shapes fall under four main categories.
1. Piecewise linear functions
   a. Triangular MF
   b. Trapezoidal MF

2. Gaussian distribution function
   a. Simple Gaussian MF
   b. Two sided Composite Gaussian MF

3. Bell shaped function

4. Sigmoidal function
   a. Product sigmoid MF
   b. Difference sigmoid MF

5. Polynomial based functions
   a. Polynomial-Z-MF
   b. Polynomial-S-MF
   c. Polynomial-PI-MF

The “shape” of the MF is an important criterion that has to be considered. The simplest and the most commonly used MF is triangular type, which can be symmetrical or asymmetrical in shape. A trapezoidal MF has the shape of a truncated triangle. Both the Gaussian and bell functions are smooth and non-zero at all the points. In practice, one or two types of MF are sufficient to solve most of the problems. MF which is one of the most important criteria for fuzzy logic conversion are represented by mathematical calculus and the functions which are segmented with straight lines and tables.

MF is associated with each linguistic label which consists of two inputs and one output response associated with it. The linguistic labels are
divided into seven groups. They are, NB-Negative Big; NM-Negative Medium; NS- Negative Small; Z-zero; PS-Positive Small; PM-Positive medium; PB-Positive Big.

Each of the inputs and the output contain MF with all these seven linguistics. The Figure 5.5(a) shows the torque error, Figure 5.5(b) shows the change in torque error and Figure 5.5(c) shows the torque limit of the FLC.

The relationship between fuzzy inputs and the fuzzy outputs is derived with the help of fuzzy rule base as given in Table 5.1.

<table>
<thead>
<tr>
<th>Δe</th>
<th>e</th>
<th>NB</th>
<th>NM</th>
<th>NS</th>
<th>Z</th>
<th>PS</th>
<th>PM</th>
<th>PB</th>
</tr>
</thead>
<tbody>
<tr>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td></td>
</tr>
<tr>
<td>NM</td>
<td>NB</td>
<td>NB</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td></td>
</tr>
<tr>
<td>NS</td>
<td>NB</td>
<td>NM</td>
<td>NS</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td></td>
</tr>
<tr>
<td>Z</td>
<td>NM</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td></td>
</tr>
<tr>
<td>PS</td>
<td>NM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PS</td>
<td>PM</td>
<td>PB</td>
<td></td>
</tr>
<tr>
<td>PM</td>
<td>NS</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td></td>
</tr>
<tr>
<td>PB</td>
<td>Z</td>
<td>PS</td>
<td>PM</td>
<td>PM</td>
<td>PB</td>
<td>PB</td>
<td>PB</td>
<td></td>
</tr>
</tbody>
</table>

The d, q projections of the stator phase currents are then compared to their reference values \(i_{sqref}\) and \(i_{sdrf}\) and corrected by means of PI current controllers. The outputs of the current controllers are passed through the inverse Park transform and a new stator voltage vector is impressed to the motor using the HPWM/SVPWM technique.
Figure 5.5 Inputs and Outputs of FLC (a) Torque Error Input (b) Change in Torque Error Input (c) Torque Limit Output
5.3 RESULTS OF THE SIMULATION AND DISCUSSION

The reference load torque of 8 Nm is applied to the PMSM motor and the results are studied for FLC with HPWM and FLC with SVPWM based FOC of PMSM. The parameters of PMSM used in the simulation (Table 3.4. Parameters of PMSM) in chapter 3 are used here for simulation in SIMULINK under MATLAB.

In order to quantitatively evaluate the effectiveness of the proposed system for the torque pulsation minimization, the torque ripple factors which are introduced in chapter 3 Equation (3.45) is used in this chapter. The controller parameters of the proposed FLC with HPWM and FLC with SVPWM based FOC of PMSM is shown in Table 5.2.

Table 5.2 Controller Parameters of FLC Based FOC of PMSM

<table>
<thead>
<tr>
<th>Controller parameters</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>PI-I,q current controller</td>
<td>K_p=10, K_i=0.1</td>
</tr>
<tr>
<td>PI-I,d current controller</td>
<td>K_P=7, K_i=0.1</td>
</tr>
<tr>
<td>PI-torque controller</td>
<td>K_p=0.36, K_i=5.4</td>
</tr>
<tr>
<td>Hysteresis current band limit</td>
<td>± 0.1</td>
</tr>
</tbody>
</table>

The simulink model of FLC with HPWM based FOC of PMSM is shown in Figure 5.6 and that of HPWM pulse production is shown in Figure 5.7. The Hysteresis band selected for simulation is h=0.1. Once, the torque waveform reaches steady state of given reference torque of 8 Nm, it oscillates around the reference torque. The Figure 5.8 shows the output torque waveform of FLC with HPWM based FOC of PMSM, which oscillates around the reference torque of 8 Nm. The Figure 5.8(a) is zoomed to view the
torque ripples of FLC with HPWM based FOC of PMSM, and it is shown in Figure 5.8(b).

Figure 5.6 Simulink Model of FLC with HPWM Based FOC of PMSM

Figure 5.7 Simulink Model of HPWM Pulse Production
Figure 5.8  Torque Output Waveforms of FLC with HPWM Based FOC of PMSM (a) Torque Output (b) Torque Ripples
For the Figure 5.8(b), the torque ripple factor of the proposed PMSM driven by FOC using FLC and HPWM scheme is as per Equation (3.45) and is given below.

\[
\text{Torque Ripple Factor (TRF \%)} = \frac{8.62 - 7.1}{8} = 19\%
\]

Similarly, the simulink model of FLC with SVPWM based FOC of PMSM is shown in Figure 5.9 and that of SVPWM pulse production is shown in Figure 5.10.

![Simulink Model of FLC with SVPWM Based FOC of PMSM](image-url)

**Figure 5.9 Simulink Model of FLC with SVPWM Based FOC of PMSM**
Figure 5.10 Simulink Model of SVPWM Pulse Production

The results of the simulation of FOC based PMSM with FLC and SVPWM is shown in the Figures 5.11, 5.12 and 5.13.

Figure 5.11 Torque Output waveform of FLC with SVPWM Based FOC of PMSM
For Figure 5.12, the torque ripple factor of the proposed PMSM driven by FOC using FLC and SVPWM scheme is as per equation (3.45) and is given below.

\[
\text{Torque Ripple Factor (TRF \%)} = \frac{8.05 - 7.91}{8} = 1.75\%
\]
The comparative results of field oriented control of PMSM using both HPWM and SVPWM techniques and is shown in Table 5.3.

**Table 5.3 Results of the Comparison of FLC based FOC of PMSM**

<table>
<thead>
<tr>
<th>FOC Method</th>
<th>Torque Ripple Factor (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FLC with HPWM</td>
<td>19</td>
</tr>
<tr>
<td>FLC with SVPWM</td>
<td>1.75</td>
</tr>
</tbody>
</table>

From the waveforms it is inferred that the torque ripples in the case of FLC with HPWM is 19% and the ripple is 1.75 % by SVPWM technique. Thus, by SVPWM, the ripples are reduced completely.

### 5.4 HARDWARE IMPLEMENTATION OF FLC WITH SVPWM IN FPGA

The hardware comprises of PMSM with eddy current loading, Intelligent Power module (IPM), Spartan 3A DSP FPGA controller, three phase auto transformer, the user manual of the IPM module is given in Appendix 3. The inbuilt current sensor of the IPM senses the currents $i_a$, $i_b$ and $i_c$ and is given to the FPGA controller. In the FPGA controller, the control technique is implemented and the outputs are recorded using Power Analyser and Mixed signal Oscilloscope. The Figure 5.14 shows the hardware block realization of FLC with SVPWM based FOC of PMSM using FPGA Spartan-3A DSP controller. The Figure 5.15 shows the hardware block diagram of FLC with SVPWM based Field oriented Control of PMSM using FPGA controller.
Figure 5.14 Hardware Block Realization of FLC with SVPWM Based FOC of PMSM Using FPGA Spartan-3A DSP Controller

Figure 5.15 Hardware Block Diagram of FLC with SVPWM Based FOC of PMSM Using FPGA Controller
5.4.1 Field Programmable Gate Array (FPGA)

FPGA is low cost, high-performance DSP solution for high-volume, cost-conscious applications. Their efficiency in the concurrent applications is achieved by using multiple parallel processing blocks. The processing path of the FPGA has parallel operations and therefore could produce faster and multiple control loops in a single FPGA unit itself.

One of the unique features of FPGA is that it could reconfigure the FPGA so that it could produce better flexibility. It distributes memory throughout the device, so the dedicated memory needed by each task is permanently allocated. This provides a high degree of isolation between tasks. The designer of an FPGA controller has complete flexibility to select any combination of peripherals and controllers.

5.4.1.1 Benefits of FPGA Controller

- Very low cost, high-performance DSP solution for high-volume, cost-conscious applications
- Their efficiency in concurrent applications is achieved by using multiple parallel processing blocks. As the processing paths are parallel in nature, it could execute different operations to compete for the same processing resources.
- The operation speeds are faster and could have multiple control loops to run a single unit of FPGA.
- Greater flexibility.
- FPGAs distribute memory throughout the device, so the dedicated memory needed by each task is permanently
allocated. This provides a high degree of isolation between tasks.

- The designer of an FPGA controller has complete flexibility to select any combination of peripherals and controllers.

5.4.1.2 Spartan-3A DSP Trainer Kit

The Spartan-3A DSP Trainer kit includes the following components and features:

1. Xilinx XC3SD1800A-FG676-4 (The Spartan-3A DSP FPGA Coding is given in Appendix 2)
2. 8 digital input using dip switches
3. 16 digital outputs using discrete LEDs
4. One Reset switch
5. FPGA configuration through
   (i) JTAG port
   (ii) Onboard Flash Prom XCF16PV048
6. On board programmable oscillator from 3MHz to 100 MHz
7. 16 × 2 LCD interface
8. ADC & DAC interface
   (i) Add on card VSDA - 03 for ADC & DAC with SDA bus
9. RS232 Serial Port
10. Pulse Width Modulation
11. USB 2.0 complaint interface (480 Mbits/sec)
The Spartan 3A DSP trainer kit is shown in the Figure 5.16.

![Figure 5.16 Spartan 3A DSP Trainer Kit](image)

5.4.2 PMSM and PI Controllers

The Parameters of Permanent magnet synchronous motor used for hardware Implementation is shown in Table 5.4 and its detailed datasheet of PMSM is given in Appendix 1.

The PI controllers and tuning are already discussed in chapter 3, is used in this chapter also. The PI-\( I_q \) current controller parameters are \( K_p=0.1 \), \( K_i=0.01 \). The PI-\( I_d \) current controller parameters \( K_p=0.1 \), \( K_i=0.01 \).

Table 5.4 Parameters of PMSM Used for Hardware Implementation

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated Power</td>
<td>1.1HP</td>
</tr>
<tr>
<td>Rated Torque</td>
<td>2.2Nm</td>
</tr>
<tr>
<td>Rated Voltage</td>
<td>220 V</td>
</tr>
<tr>
<td>Rated Current</td>
<td>3.69A</td>
</tr>
<tr>
<td>Rated Speed</td>
<td>4600 rpm</td>
</tr>
<tr>
<td>Torque Constant</td>
<td>0.6 Nm/Arms</td>
</tr>
<tr>
<td>Terminal To Terminal Resistance</td>
<td>3.07Ω</td>
</tr>
<tr>
<td>Terminal To Terminal Inductance</td>
<td>6.57 mH</td>
</tr>
<tr>
<td>Moment of Inertia</td>
<td>1.8 kgcm(^2)</td>
</tr>
</tbody>
</table>
5.4.3 Fuzzification and Defuzzification in FPGA

The inputs to the FLC are Torque error and change in torque error and the output is torque limit which are described in the Figure 5.17.

Figure 5.17 (a) Torque Error Input to FLC, (b) Change in Torque Error Input to FLC, (c) Torque Limit Output of FLC
5.4.4 Program Realization Using Flowchart

The flowchart of FLC with SVPWM based FOC of PMSM using FPGA Spartan-3A DSP controller is shown in Figure 5.18. The Figure 5.19 shows the Snapshot of overall hardware setup of FLC with SVPWM based FOC of PMSM using FPGA Spartan-3A DSP controller.

![Flowchart](image.png)

Figure 5.18 Flowchart of FLC with SVPWM Based FOC of PMSM Using FPGA Spartan-3A DSP Controller
5.4.5 Results of Real Time Implementation and Discussion

The parameters of PMSM are shown in Table 5.4 used for real time implementation. The Figures 5.20 and 5.21 shows the SVPWM pulses and symmetrical switching pattern timing of SVPWM.
Figure 5.20 SVPWM Pulses

Figure 5.21 Symmetrical Switching Pattern Timing of SVPWM
The Figures 5.22 shows the three phase line to line voltages (Vab, Vbc, Vca) and corresponding current waveforms (Iab, Ibc, Ica) without filtering. The Figures 5.23 shows the voltage and current Total Harmonic Distortion (THD) datas of the Figure 5.22. From the Figure 5.23 the THD % of V_{ab} = 12.19, I_{ab} = 4.77, V_{bc} = 12.69, I_{bc} = 3.99, V_{ca} = 12.17 and I_{ca} = 4.16.

Figure 5.22 Three Phase Line to Line Voltages and Corresponding Current Waveforms
If tuning of the PI controllers are not properly done, it will increase the Voltage and Current THD. The increase in Voltage and Current THD will increase the torque ripples.

The Figures 5.24 and 5.25 shows Inverter Line to line Voltage waveform and current waveform at 75% loading. The Figures 5.26 and 5.30 show the torque waveform for 75% and 100% loading.
Figure 5.24 Inverter Line to line Voltage Waveform for 75% Loading

Figure 5.25 Current Waveform for 75% Loading
Figure 5.26 Torque Waveform for 75% loading

From the Figure 5.26 the motor torque (Tm) follows the reference motor torque (Tref) with less torque ripples and it is shown in the circled portion of waveform which is zoomed and shown inside the Figure 5.26. The torque ripple factor for the proposed scheme as per equation (3.45) is given below for Figure 5.26,

\[
\text{Torque Ripple Factor (TRF \%)} = \frac{1.65 - 1.62}{1.65} = 1.81\%
\]

The Figures 5.27 and 5.28 shows Inverter Line to line Voltage waveform and current waveform at 100% loading.
Figure 5.27 Inverter Line to line Voltage Waveform for 100% Loading

Figure 5.28 Current Waveform for 100% Loading
From the Figure 5.29 the motor torque (Tm) follows the reference motor torque (Tref) with less torque ripples and it is shown in the circled portion of waveform which is zoomed and shown inside the Figure 5.29.

The torque ripple factor for the proposed scheme as per Equation (3.45) is given below for Figure 5.29.

\[
\text{Torque Ripple Factor (TRF \%)} = \frac{2.24 - 2.2}{2.2} = 1.81\%
\]

It is clear that the variation in Torque ripples is less in case of FLC and they can achieve a minimum torque ripple. It has been viewed that the proposed control strategy has helped in reducing the torque ripples to 1.81%. Thus, by using FLC based controller, ripples are reduced to a great extent.

The dynamic torque waveform is shown in the Figure 5.30. The reference motor torque (Tref) is varied to 25, 50, 75 and 100% with specified
interval of time and it is seen that the motor torque (Tm) follows the reference motor torque (Tref) with less torque ripples.

Figure 5.30 Dynamic Torque Waveform

FLC based Torque controller of PMSM motor drive have been modeled and implemented using FPGA and the results have been presented to demonstrate the proposed FLC based control. The hardware results of FLC have shown the improved performance over the PI Torque controller in reducing the torque ripples to 1.81%. Hence, it can be effectively used in place of PI Torque controller.

5.5 SUMMARY

The percentage of torque ripples obtained by each of the proposed methods of FOC of PMSM drive systems are shown in Table 5.5. The Torque Ripple Factor (TRF) of PI controller is 48 %, by ILC with HPWM it is 20 %, by FLC With HPWM it is 19%, by ANN Controller With SVPWM it is 2.75%, by ILC with SVPWM it is 2 %, and by FLC With SVPWM it is 1.75%. Thus the proposed FLC with SVPWM method shows better torque reduction than the other proposed methods and all the conventional methods.
Table 5.5 TRF of the Proposed FOC Methods of PMSM Drives

<table>
<thead>
<tr>
<th>Method-FOC of PMSM</th>
<th>Torque Ripple Factor (TRF) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PI Controller</td>
<td>48%</td>
</tr>
<tr>
<td>ILC With HPWM</td>
<td>20%</td>
</tr>
<tr>
<td>FLC With HPWM</td>
<td>19%</td>
</tr>
<tr>
<td>ANN Controller With SVPWM</td>
<td>2.75%</td>
</tr>
<tr>
<td>ILC With SVPWM</td>
<td>2%</td>
</tr>
<tr>
<td>FLC With SVPWM</td>
<td>1.75%</td>
</tr>
</tbody>
</table>

The Figure 5.31 shows the comparison of torque ripple factor of the proposed methods. The result of simulation shows that the proposed FLC with SVPWM based FOC of PMSM has lesser torque ripples of 1.75%.

![Comparison of Torque ripple factor (TRF %) of Proposed Methods](image)

**Figure 5.31 Comparison of Torque Ripple Factor of the Proposed Methods**

The Table 5.6 shows the comparison of the effectiveness of the proposed FLC with SVPWM based FOC of PMSM is better than the control strategies reported earlier in the literature by others.
Table 5.6 TRF of Proposed Method and Reported TRF

<table>
<thead>
<tr>
<th>Control strategies of PMSM</th>
<th>Torque Ripple Factor (TRF) (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qian et al (2004)- Iterative Learning Control</td>
<td>3.9</td>
</tr>
<tr>
<td>Mattavelli et al (2005)- Repetitive current control</td>
<td>3.8</td>
</tr>
<tr>
<td>Gulez et al (2007a)- Novel DTC</td>
<td>7.5</td>
</tr>
<tr>
<td>H. Hasanien (2010)- Digital observer controller</td>
<td>12</td>
</tr>
<tr>
<td>Tarnik and Murgas (2011)- Additional Adaptive controller</td>
<td>4</td>
</tr>
<tr>
<td>Proposed- FLC With SVPWM based FOC</td>
<td>1.81</td>
</tr>
<tr>
<td></td>
<td>(Experimental Verification)</td>
</tr>
</tbody>
</table>

The Figure 5.32 shows the comparison of torque ripple factor of the proposed FLC with SVPWM based FOC of PMSM is better than the control strategies reported earlier in the literature by others.