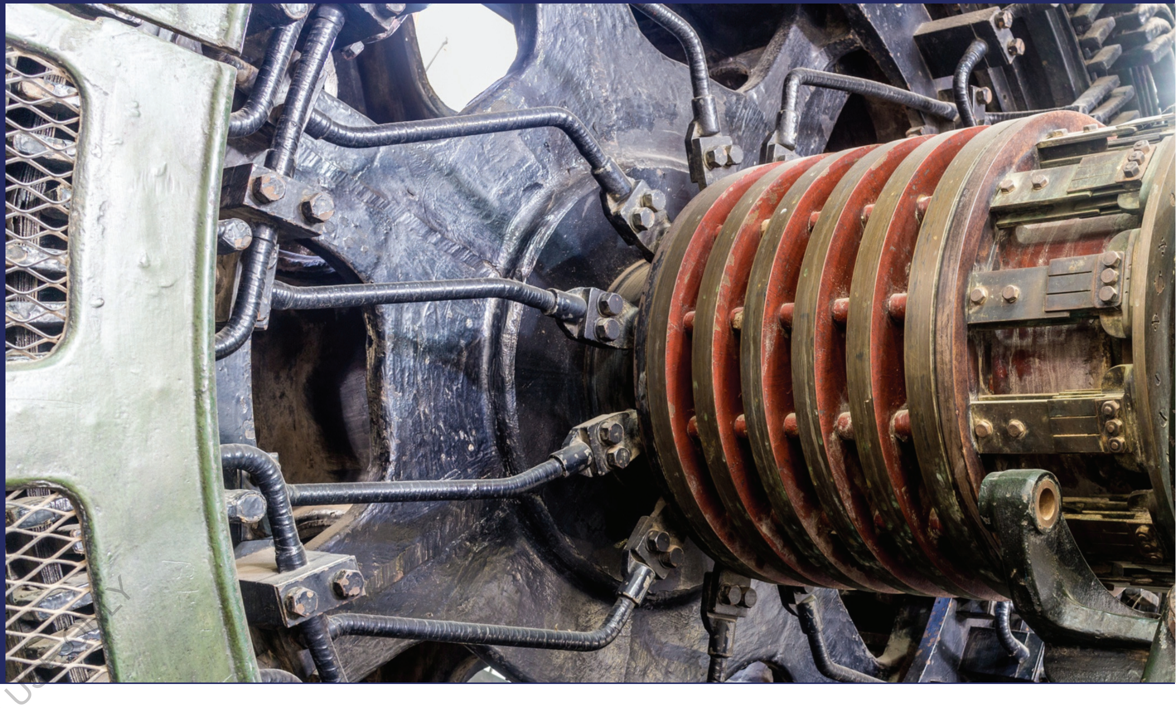
This work presents an innovative control approach for Power Factor Correction (PFC) in a three-phase PWM AC chopper. It employs the HBCC technique to drive a three-phase squirrel cage Induction Motor (IM) and supports both soft starting and speed control modes. What sets this strategy apart is its remarkable simplicity, reliability, high efficiency, and cost-effectiveness, achieved by minimizing the number of power semiconductor switches in the power circuitry. The three-phase PWM AC chopper is built using only four IGBTs, and a new closed-loop control methodology is devised, utilizing just two gate pulses to manage these four IGBTs.



Dr. Raja Reddy Duvvuru, *Associate Professor, Department of EEE, Malla Reddy Engineering College, Hyderabad.*
Dr. V.Madhusudhan Reddy, *Professor, Department of EEE, Malla Reddy Engineering College, Hyderabad.*

9 786207 456284

Raja Reddy Duvvuru
V. Madhusudhan Reddy

Induction Motor Drive with H-Bridge Cascaded Converter

LAP
LAMBERT
Academic Publishing

Raja Reddy Duvvuru
V. Madhusudhan Reddy

Induction Motor Drive with H-Bridge Cascaded Converter

Raja Reddy Duvvuru
V. Madhusudhan Reddy

Induction Motor Drive with H-Bridge Cascaded Converter

LAP LAMBERT Academic Publishing

Imprint

Cover image: www.ingimage.com

Publisher:
LAP LAMBERT Academic Publishing
is a trademark of
Dodo Books Indian Ocean Ltd. and OmniScriptum S.R.L publishing group

120 High Road, East Finchley, London, N2 9ED, United Kingdom
Str. Armeneasca 28/1, office 1, Chisinau MD-2012, Republic of Moldova, Europe
Printed at: see last page
ISBN: 978-620-7-45628-4

TABLE OF CONTENTS

CHAPTER – 1

INTRODUCTION

1.1 Introduction

AC voltage regulators, also called as AC voltage controllers, are used in various applications that require a regulated AC voltage. Lighting control using dimmer circuits, domestic and industrial heating, speed control and soft starters for the induction motors are examples of such applications [1], [2]. Different topologies with different control methods of these regulators in single phase applications and also in three phase applications are presented. The purpose of AC voltage controller is to vary the root mean square (RMS) value of its output that applied to the load circuit. There are three control methods are offered to achieve this objective; ON/OFF method, phase angle (PA) method and pulse width modulation (PWM) method. All three control methods can be implemented in both single-phase and three-phase applications.

In ON/OFF control method, thyristors (i.e. Silicon Controlled Rectifiers) are used as power switches to connect/disconnect the circuit of the load to/from the AC voltage source continuously. Connection is occurred for a few integral cycles and disconnection for the next few cycles of the feeding voltage. Adjusting the number of conducted and interrupted cycle’s controls the RMS magnitude of the output voltage. In ON/OFF method, the generated harmonics by the switching actions are reduced as silicon-controlled .rectifiers (SCRs) are switched ON at zero voltage and switched OFF at zero current. However, undesirable sub-harmonic components may be produced [3].

Applications of this method are restricted to heating and temperature control systems due to the discontinuity of the power source at low demand levels. In PA control method, the output of the AC voltage controller is regulated by adjusting the firing angles of SCRs. The power circuit of a single-phase regulator with PA control method is generally consisted of two thyristors which are joined back to back between the AC source and the load circuit, while three-phase regulator is composed of three pairs of SCRs. In [4], [5], soft starting for induction motor (IM) fed from a thyristorized voltage regulator is presented. The artificial techniques are utilized to adjust the motor voltage by varying the firing angles of the thyristors at certain operating instant of speed and torque commands. In [6], a voltage ramp technique is presented for starting of an AC motor. The voltage, in ramp technique, is increased gradually by adjusting the SCRs firing angles during starting of the motor. In [7], [8], a closed loop current control approach that determines the firing angles of thyristors required to keep the motor current at starting instant within a limit value is presented. In these approaches, a smooth start-up of the IM is obtained. However, numerous sensors and zero crossing detection (ZCD) circuits are required which make these controllers are complicated and expensive. In addition, the thyristorized AC voltage controller provides significant harmonics and low input power factor (PF) even if the load is a pure resistive. Recent developments in semiconductor switches make it possible to replace SCRs by modern power semiconductor switches like MOSFETs and IGBTs. Using PWM control method with the modern power switches, the AC voltage regulators performance can be enhanced in terms of harmonics, filter size, input PF and voltage control range [9]–[11]. In [12], a speed control of two-phase IM fed from single phase PWM AC chopper is presented.

POWER FACTOR is the ratio between the useful (true) power (kW) to the total (apparent) power (kVA) consumed by an item of a.c. electrical equipment or a complete electrical installation. It is a measure of how efficiently electrical power is converted into useful work output. The ideal power factor is unity, or one. Anything less than one means that extra power is required to achieve the actual task at hand. All current flow causes losses both in the supply and distribution system. A load with a power factor of 1.0 results in the most efficient loading of the supply. A load with a power factor of, say, 0.8, results in much higher losses in the supply system and a higher bill for the consumer. A comparatively small improvement in power factor can bring about a significant reduction in losses since losses are proportional to the square of the current. When the power factor is less than one the ‘missing’ power is known as reactive power which unfortunately is necessary to provide a magnetising field required by motors and other inductive loads to perform their desired functions. Reactive power can also be interpreted as wattless, magnetising or wasted power and it represents an extra burden on the electricity supply system and on the consumer’s bill. A poor power factor is usually the result of a significant phase difference between the voltage and current at the load terminals, or it can be due to a high harmonic content or a distorted current waveform. A poor power factor is generally the result of an inductive load such as an induction motor, a power transformer, a ballast in a luminaire, a welding set or an induction furnace. A distorted current waveform can be the result of a rectifier, an inverter, a variable speed drive, a switched mode power supply, discharge lighting or other electronic loads. A poor power factor due to inductive loads can be improved by the addition of power factor correction equipment, but a poor power factor due to a distorted current waveform requires a change in equipment design or the addition of harmonic filters. Some inverters are quoted as having a power factor of better than 0.95 when, in reality, the true power factor is between 0.5 and 0.75. The figure of 0.95 is based on the cosine of the angle between the voltage and current but does not take into account that the current waveform is discontinuous and therefore contributes to increased losses. An inductive load requires a magnetic field to operate and in creating such a magnetic field causes the current to be out of phase with the voltage (the current lags the voltage). Power factor correction is the process of compensating for the lagging current by creating a leading current by connecting capacitors to the supply. A sufficient capacitance is connected so that the power factor is adjusted to be as close to unity as possible.

Power factor explained Consider a single-phase induction motor. If the motor presented a purely resistive load to the supply, the current flowing would be in-phase with the voltage. This is not the case. The motor has a magnet and the magnetizing current is not in phase with the voltage. The magnetizing current is the current that establishes the flux in the iron and, being out of phase, causes the shaft of the motor to rotate. The magnetizing current is independent of the load on the motor and will typically be between 20% and 60% of the rated full load current of the motor. The magnetizing current does not contribute to the work output of the motor.

The power factor can be expressed in two ways:

1. Power factor (pf) = Useful power (kW) divided by the total power (kVA),
2. Power factor (pf) = The cosine of the angle between useful power and total power = cos ø.

REASONS TO IMPROVE PF:

The benefits that can be achieved by applying the correct power factor correction are:

1. Environmental benefit.
2. Reduction of power consumption due to improved energy efficiency.
3. Reduced power consumption means less greenhouse gas emissions and fossil fuel depletion by power stations.
4. Reduction of electricity bills
5. Extra kVA available from the existing supply Reduction of I2 R losses in transformers and distribution equipment
6. Reduction of voltage drops in long cables.
7. Extended equipment life – Reduced electrical burden on cables and electrical components.

Power factor correction is the term given to a technology that has been used since the turn of the 20th century to restore the power factor to as close to unity as is economically viable. This is normally achieved by the addition of capacitors to the electrical network which compensate for the reactive power demand of the inductive load and thus reduce the burden on the supply. There should be no effect on the operation of the equipment. To reduce losses in the distribution system, and to reduce the electricity bill, power factor correction, usually in the form of capacitors, is added to neutralize as much of the magnetizing current as possible.

Capacitors contained in most power factor correction equipment draw current that leads the voltage, thus producing a leading power factor. If capacitors are connected to a circuit that operates at a nominally lagging power factor, the extent that the circuit lags is reduced proportionately. Typically the corrected power factor will be 0.92 to 0.95. Some power distributors offer incentives for operating with a power factor of better than 0.9, for example, and some penalize consumers with a poor power factor. There are many ways that this is metered but the net result is that in order to reduce wasted energy in the distribution system, the consumer is encouraged to apply power factor correction. Most Network Operating companies now penalize for power factors below 0.95 or 0.9.

HOW TO INCREASE PF:

Power factor correction is achieved by the addition of capacitors in parallel with the connected motor or lighting circuits and can be applied at the equipment, distribution board or at the origin of the installation. Static power factor correction can be applied at each individual motor by connecting the correction capacitors to the motor starter. A disadvantage can occur when the load on the motor changes and can result in under or over correction. Static power factor correction must not be applied at the output of a variable speed drive, solid state soft starter or inverter as the capacitors can cause serious damage to the electronic components. Over-correction should not occur if the power factor correction is correctly sized. Typically the power factor correction for an individual motor is based on the non load (magnetizing) power since the reactive load of a motor is comparatively constant compared to actual kW load over compensation should be avoided. Care should be taken when applying power factor correction star/delta type control so that the capacitors are not subjected to rapid on-off-on conditions. Typically the correction would be placed on either the Main or Delta contactor circuits. Power factor correction applied at the origin of the installation consists of a controller monitoring the VAr's and this controller switches capacitors in or out to maintain the power factor better than a preset limit (typically 0.95). Where 'bulk' power factor correction is installed, other loads can in theory be connected anywhere on the network.

CHAPTER – 2

LITERATURE REVIEW

2.1 SYSTEM DESCRIPTION AND OPERATION PRINCIPLE

Fig. 1 freewheeling stage, the power switch represents a schematic diagram of the proposed three phase PWM AC chopper fed an IM. The chopper is composed only of four power electronics switches (S1, S2, S3 and S4) that are illustrated in the figure. The three power switches (S1, S2 and S3) are series-connected with the motor. While, the power switch (S4) is parallel-connected via a poly phase bridge rectifier with the motor. The series-connected switches are utilized to continuously connect and disconnect the motor to and from the AC supply, respectively. Hence, they regulate the delivered power to the motor. While the parallel-connected switch (S4) offers a freewheeling way for discharging the energy kept in the motor windings when the series-connected switches are turned OFF. As series and parallel devices operate in a complementary way, a dead time is introduced to avoid the commutation problems. There are three operating stages: active, freewheeling and dead time.

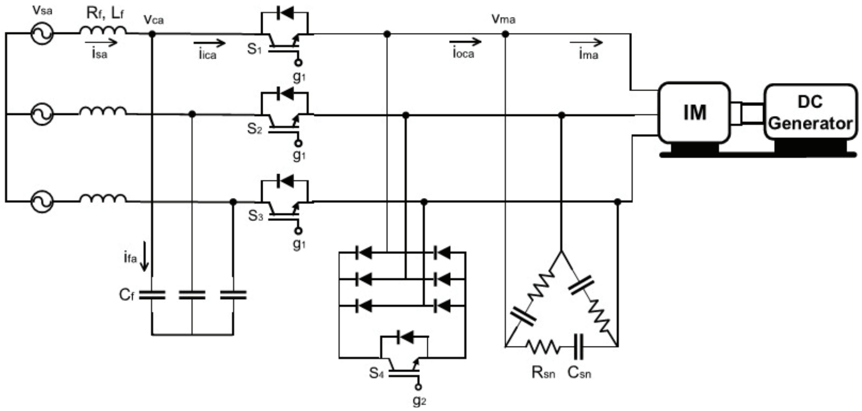


Fig.2.1 Schematic proposed diagram-Pulse Width Modulation AC Chopper

In dead time period, all four devices are turned OFF. The currents paths of the proposed PWM AC chopper fed IM in its three operating stages are illustrated by Fig. 2. A three phase 1-connected snubber circuit, which has a resistance R_{sn} and a capacitance C_{sn} per phase, is used to minimize high voltage spikes at IM terminals due to switching of the chopper as well as providing the current path of IM during the dead time period. The input filter is composed of three inductors (whose resistance is R f and inductance is Lf per phase) and Y-connected three capacitors (whose capacitance C_f per phase). The LC input filter is used with the proposed PFC technique in order to reduce the harmonics of the supply current due to switching of the chopper. The proposed control circuit only generates two PWM complementary gate pulses (g1 and g2) which are used to drive the chopper IGBTs in order to provide the three main tasks of the proposed control strategy.

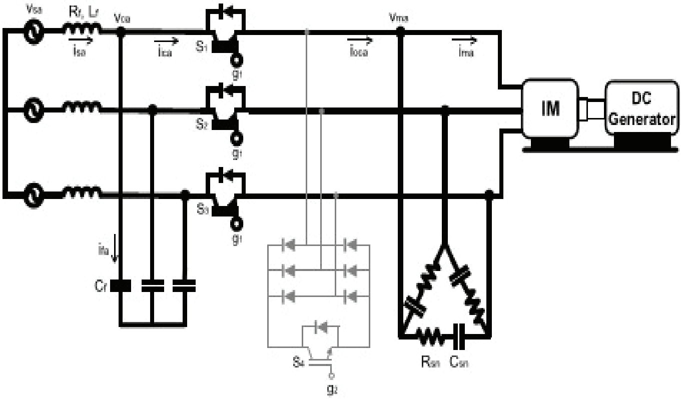


Fig.2.1 (a) Active stage

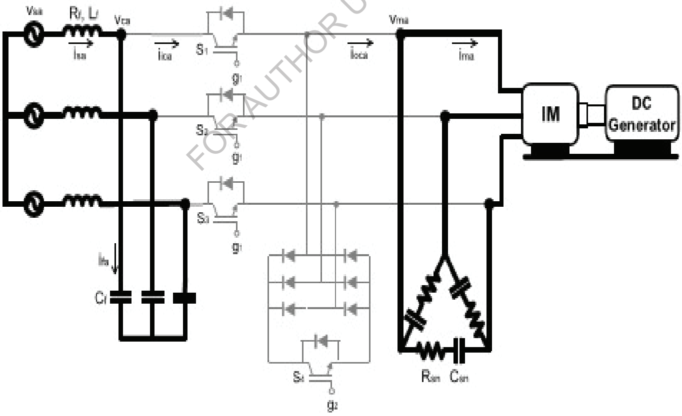


Fig. 2.1 (b) Dead time stage

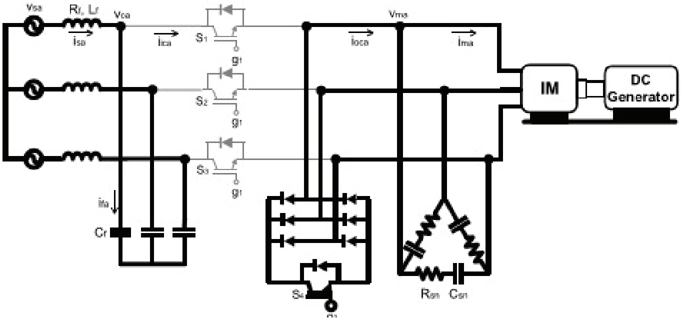


Fig.2.1 (c) Freewheeling stage

FIGURE 2.1The currents paths of the proposed PWM AC chopper fed IM. (a) Active stage. (b) Dead time stage. (c) Freewheeling stage.

Induction Motor

An induction motor (IM) is a type of asynchronous AC motor where power is supplied to the rotating device by means of electromagnetic induction. Other commonly used name is squirrel cage motor due to the fact that the rotor bars with short circuit rings resemble a squirrel cage (hamster wheel).An electric motor converts electrical power to mechanical power in its rotor.

There are several ways to supply power to the rotor. In a DC motor this power is supplied to the armature directly from a DC source, while in an induction motor this power is induced in the rotating device.

An induction motor is sometimes called a rotating transformer because the stator (stationary part) is essentially the primary side of the transformer and the rotor (rotating part) is the secondary side. Induction motors are widely used, especially polyphase induction motors, which are frequently used in industrial drives.

The Induction motor is a three phase AC motor and is the most widely used machine. Its characteristic features are-

- Simple and rugged construction
- Low cost and minimum maintenance
- High reliability and sufficiently high efficiency
- Needs no extra starting motor and need not be synchronized
- An Induction motor has basically two parts – Stator and Rotor

The Stator is made up of a number of stampings with slots to carry three phase windings. It is wound for a definite number of poles. The windings are geometrically spaced 120 degrees apart. Two types of rotors are used in Induction motors - Squirrel-cage rotor and Wound rotor.

2.2 INDUCTION MOTOR GENERAL PRINCIPLE

As a general rule, conversion of electrical power into mechanical power takes place in the rotating parts of an electrical motor. In dc motor, the electrical power is conducted directly in armature the rotating part of the motor through brush or commutates and hence dc motor called as conduction motor but in case of induction motor the motor does not receive the electrical power by conduction but by induction in exactly same way as the secondary of a 2-winding transformer receives its power from the primary. That is why such motor known as induction motor.

In fact, an induction motor can be treated as a rotating transformer i.e. one in which primary winding is stationary but the secondary is free to rotate. Of all the a.c. motors, the poly phase induction motor is the one which is extensively used for various kinds of industrial drives.

When a three-phase supply is connected to the stator windings, a rotating magnetic field is produced. As the magnetic flux cuts a bar on the rotor, an e.m.f. is induced in it and since it is joined, via the end conducting rings, to another bar one pole pitch away, current flows in the bars.

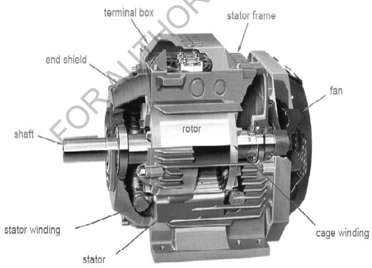


Fig.2.2 Showing various parts of induction motor

The magnetic field associated with this current flowing in the bars interacts with the rotating magnetic field and a force is produced, tending to turn the rotor in the same direction as the rotating magnetic field. Similar forces are applied to all the conductors on the rotor, so that a torque is produced causing the rotor to rotate.

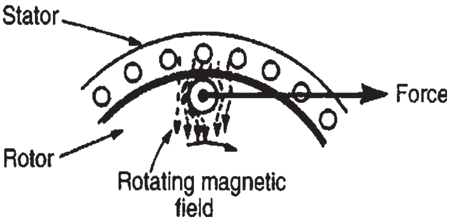


Fig.2.3 Showing production of magnetic field

They are widely used for different applications ranging from small induction motors in washing machines, household fans etc to vary large induction motors which are capable of tens of thousands of kW in output, for pipeline compressors, wind-tunnel drives and overland conveyor systems. Through electromagnetic induction, the rotating magnetic field induces a current in the conductors in the rotor, which in turn sets up a counterbalancing magnetic field that causes the rotor to turn in the direction the field is rotating. The rotor must always rotate slower than the rotating magnetic field produced by the polyphase electrical supply; otherwise, no counterbalancing field will be produced in the rotor. Induction motors are the workhorses of industry and motors up to about 500 kW (670 horsepower) in output are produced in highly standardized frame sizes, making them nearly completely interchangeable between manufacturers.

2.3 CONSTRUCTION OF INDUCTION MOTOR

An induction motor consists of many parts, the stator and rotor being the basic subsystems of the machine. An exploded view of a squirrel-cage motor is shown .The motor case (frame), ribbed outside for better cooling, houses the stator core with a three phase winding placed in slots on the periphery of the core. The stator core is made of thin (0.3 mm to 0.5 mm) soft-iron laminations, which are stacked and screwed together. Individual laminations are covered on both sides with insulating lacquer to reduce eddy-current losses. On the front side, the stator housing is closed by a cover, which also serves as a support for the front bearing of the rotor. Usually, the cover has drip-proof air intakes to improve cooling. The stator is the stationary electrical part of the motor. The stator core of a NEMA motor is made up of several hundred thin laminations.

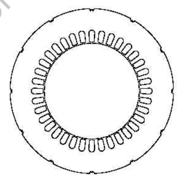


Fig.2.4 Lamination of core

Stator laminations are stacked together forming a hollow cylinder. Coils of insulated wire are inserted into slots of the stator core. Each grouping of coils, together with the steel core it surrounds form an electromagnet.

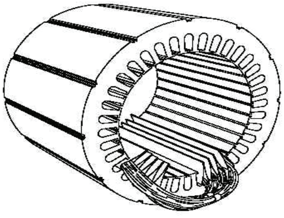


Fig.2.5 Windings in stator

Electromagnetism is the principle behind motor operation. The stator windings are connected directly to the power source. The rotor, whose core is also made of laminations, is built around a shaft, which transmits the mechanical power to the load. The rotor is equipped with cooling fins. At the back, there is another bearing and a cooling fan affixed to the rotor. The fan is enclosed by a fan cover. Access to the stator winding is provided by stator terminals located in the connection box that covers an opening in the stator housing. Open-frame, partly enclosed, and totally enclosed motors are distinguished by how well the inside of stator is sealed from the ambient air. Totally enclosed motors can work in extremely harsh environments and in explosive atmospheres, for instance, in deep mines or lumber mills. However, the cooling effectiveness suffers when the motor is tightly sealed, which reduces its power rating.

The squirrel-cage rotor winding, illustrated, consists of several bars connected at both ends by end rings. The rotor cage shown is somewhat oversimplified, practical rotor windings being made up of more than few bars, not necessarily round, and slightly skewed with respect to the longitudinal axis of the motor. In certain machines, in order to change the inductance-to-resistance ratio that strongly influences mechanical characteristics of the motor, rotors with deep-bar cages and double cages are used.

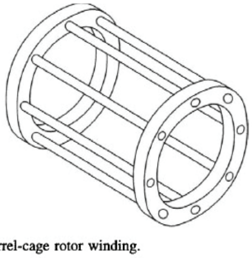


Fig 2.6 Squirrel cage rotor winding

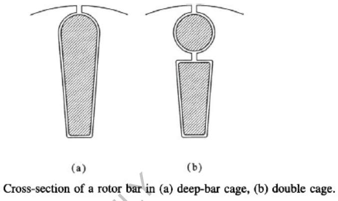


Fig 2.7 Cross-section of Rotor bar

The stator consists of wound 'poles' that carry the supply current to induce a magnetic field that penetrates the rotor. In a very simple motor, there would be a single projecting piece of the stator (a salient pole) for each pole, with windings around it; in fact, to optimize the distribution of the magnetic field, the windings are distributed in many slots located around the stator, but the magnetic field still has the same number of north-south alternations. The number of 'poles' can vary between motor types but the poles are always in pairs (i.e. 2, 4, 6, etc.).

Induction motors can be built to run on either single-phase or three-phase power. Single-phase power is more widely available in residential buildings, but cannot produce a rotating field in the motor (the field merely oscillates back and forth), so single-phase induction motors must incorporate some kind of starting mechanism to produce a rotating field. They would, using the simplified analogy of salient poles, have one salient pole per pole number; a four-pole motor would have four salient poles. Three-phase motors have three salient poles per pole number, so a four-pole motor would have twelve salient poles. This allows the motor to produce a rotating field, allowing the motor to start with no extra equipment and run more efficiently than a similar single-phase motor.

Types of three phase induction motor rotor .There are two types of induction motor rotors:

1. Squirrel-cage rotor or simply cage rotor.
2. Phase wound or wound rotors.

Squirrel cage rotor: Squirrel cage motor works on the principle of Electromagnetism. It consists of Rotor, Stator and other parts like bearings, cylindrical laminated core, shaft, etc. The function of bearings in cage rotor motor is to reduce friction between the rotating and stationary parts of the machine. The rotor of the motor consists of a cylindrical laminated core with parallel slots for carrying the rotor conductors. The rotor conductors are not wires, but it consists of heavy bars of copper, aluminum, or an alloy. The shaft is used in the motor to transfer mechanical power from or to the machine. The stator is the outer stationary part of the motor.

Application of Squirrel Cage Induction Motor

Squirrel cage induction motors are commonly used in many industrial applications. They are particularly suited for applications where the motor must maintain a constant speed, be self-starting, or there is a desire for low maintenance.

These motors are commonly used in:

- Centrifugal pumps
- Industrial drives (e.g. to run conveyor belts)
- Large blowers and fans
- Machine tools
- Lathes and other turning equipment

Advantages of Squirrel Cage Induction Motor

Some advantages of squirrel cage induction motors are:

- They are low cost
- Require less maintenance (as there are no slip rings or brushes)
- Good speed regulation (they are able to maintain a constant speed)
- High efficiency in converting electrical energy to mechanical energy (while running, not during startup)
- Have better heat regulation (i.e. don’t get as hot)
- Small and lightweight
- Explosion proof (as there are no brushes which eliminate the risks of sparking)

Disadvantages of Squirrel Cage Induction Motor

Although squirrel cage motors are very popular and have many advantages – they also have some downsides. Some disadvantages of squirrel cage induction motors are:

- Very poor speed control
- Although they are energy efficient while running at full load current, they consume a lot of energy on startup
- They are more sensitive to fluctuations in the supply voltage. When the supply voltage is reduced, induction motor draws more current. During voltage surges, increase in voltage saturates the magnetic components of the squirrel cage induction motor
- They have high starting current and poor starting torque (the starting current can be 5-9 times the full load current; the starting torque can be 1.5-2 times the full load torque)

Wound rotor or slip ring rotor: The wound rotor consists of a slotted armature. Insulated conductors are put in the slots and connected to form a three-phase double layer distributed winding similar to the stator winding. The windings of the rotor are connected in star. Rotor windings are distributed uniformly and usually connected in the star with here leads brought out of the machine by via slip rings placed on the shaft. The slip rings are tapped using copper carbon brushes. Wound rotor construction is generally used for large size machine, where the starting torque requirements are stringent. External resistance can be added in the rotor circuit through slip ring for reducing the starting current and simultaneously the starting torque.

Difference between Squirrel Cage and Slip Ring Induction Motor

While slip ring induction motors (also known as wound-rotor motor) aren't as popular as squirrel cage induction motors, they do have a few advantages.

Below is a comparison table of squirrel cage vs wound rotor type motors:

Parameter	**Squirrel Cage Motor**	**Slip Ring Motor**
Cost	Low	High
Maintenance	Low	High
Speed Control	Poor	Good
Efficiency on startup	Poor	Good
Efficiency during operation	Good	Poor
Heat regulation	Good	Poor
In rush current & torque	High	Low

2.4 PROPOSED TOPOLOGY

The proposed control strategy has three main control objectives: soft starting, speed control, and input power factor correction (PFC). This strategy is depending on the control of the applied voltage across IM terminals using AC chopper. Fig. 3 illustrates the schematic diagram of the proposed control strategy. It has two control loops. The inner control loop uses HBCC to force the chopper actual current signals to track their command current signals to achieve input PFC, whereas the outer control loop determines the magnitude of the reference currents either from starting mode or speed control mode. As a result, the inner loop controls the phase and the outer loop controls the magnitude of the chopper currents. In the first, the soft starting mode is working, and by giving a switching pulse to the selector switch, the speed control mode is activated and the soft starting mode is turned off.

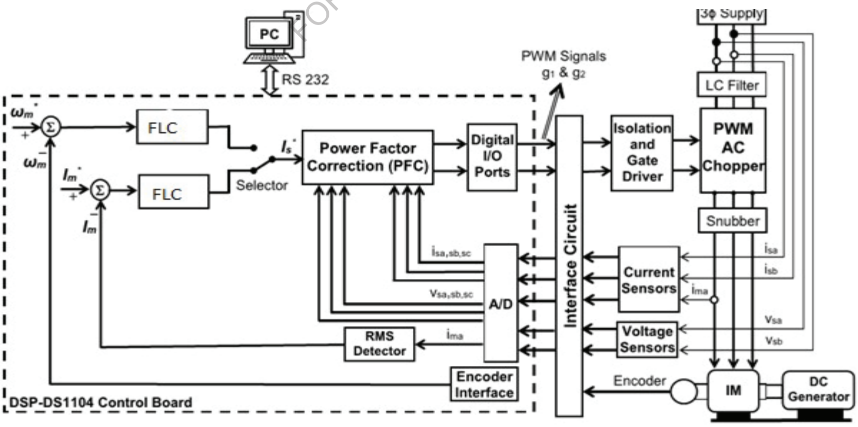


Fig.2.8 Control Scheme of Proposed topology

CHAPTER – 3

FUZZY LOGIC CONTROLLER

3.1 FUZZY LOGIC

In recent years, the number and variety of applications of fuzzy logic have increased significantly. The applications range from consumer products such as cameras, camcorders, washing machines, and microwave ovens to industrial process control, medical instrumentation, decision-support systems, and portfolio selection.

To understand why use of fuzzy logic has grown, you must first understand what is meant by fuzzy logic.

Fuzzy logic has two different meanings. In a narrow sense, fuzzy logic is a logical system, which is an extension of multivalve logic. However, in a wider sense fuzzy logic (FL) is almost synonymous with the theory of fuzzy sets, a theory which relates to classes of objects with un sharp boundaries in which membership is a matter of degree. In this perspective, fuzzy logic in its narrow sense is a branch of fl. Even in its more narrow definition, fuzzy logic differs both in concept and substance from traditional multivalve logical systems.

In fuzzy Logic Toolbox software, fuzzy logic should be interpreted as FL, that is, fuzzy logic in its wide sense. The basic ideas underlying FL are explained very clearly and insightfully in Foundations of Fuzzy Logic. What might be added is that the basic concept underlying FL is that of a linguistic variable, that is, a variable whose values are words rather than numbers. In effect, much of FL may be viewed as a methodology for computing with words rather than numbers. Although words are inherently less precise than numbers, their use is closer to human intuition. Furthermore, computing with words exploits the tolerance for imprecision and thereby lowers the cost of solution.

Another basic concept in FL, which plays a central role in most of its applications, is that of a fuzzy if-then rule or, simply, fuzzy rule. Although rule-based systems have a long history of use in Artificial Intelligence (AI), what is missing in such systems is a mechanism for dealing with fuzzy consequents and fuzzy antecedents. In fuzzy logic, this mechanism is provided by the calculus of fuzzy rules. The calculus of fuzzy rules serves as a basis for what might be called the Fuzzy Dependency and Command Language (FDCL). Although FDCL is not used explicitly in the toolbox, it is effectively one of its principal constituents. In most of the applications of fuzzy logic, a fuzzy logic solution is, in reality, a translation of a human solution into FDCL.

A trend that is growing in visibility relates to the use of fuzzy logic in combination with neuro computing and genetic algorithms. More generally, fuzzy logic, neuro-computing, and genetic algorithms may be viewed as the principal constituents of what might be called soft computing. Unlike the traditional, hard computing, soft computing accommodates the imprecision of the real world.

The guiding principle of soft computing is: Exploit the tolerance for imprecision, uncertainty, and partial truth to achieve tractability, robustness, and low solution cost. In the future, soft computing could play an increasingly important role in the conception and design of systems who’s MIQ (Machine IQ) is much higher than that of systems designed by conventional methods.

Among various combinations of methodologies in soft computing, the one that has highest visibility at this juncture is that of fuzzy logic and neuro computing, leading to neuro-fuzzy systems. Within fuzzy logic, such systems play a particularly important role in the induction of rules from observations. An effective method developed by Dr. Roger Jang for this purpose is called ANFIS (Adaptive Neuro-Fuzzy Inference System). This method is an important component of the toolbox.

The fuzzy logic toolbox is highly impressive in all respects. It makes fuzzy logic an effective tool for the conception and design of intelligent systems. The fuzzy logic toolbox is easy to master and convenient to use. And last, but not least important, it provides a reader friendly and up-to-date introduction to methodology of fuzzy logic and its wide ranging applications.

3.2 Fuzzy logic tool box

The Fuzzy Logic Toolbox extends the MATLAB technical computing environment with tools for designing systems based on fuzzy logic. Graphical user interfaces (GUIs) guide you through the steps of fuzzy inference system design. Functions are provided for many common fuzzy logic methods, including fuzzy clustering and adaptive neuro fuzzy learning.

The toolbox lets you model complex system behaviors using simple logic rules and then implements these rules in a fuzzy inference system. You can use the toolbox as a standalone fuzzy inference engine. Alternatively, you can use fuzzy inference blocks in simulink and simulate the fuzzy systems within a comprehensive model of the entire dynamic system

3.3 Working with the fuzzy logic toolbox

The Fuzzy Logic Toolbox provides GUIs to let you perform classical fuzzy system development and pattern recognition. Using the toolbox, you can develop and analyze fuzzy inference systems, develop adaptive neuro fuzzy inference systems, and perform fuzzy clustering. In addition, the toolbox provides a fuzzy controller block that you can use in Simulink to model and simulate a fuzzy logic control system. From Simulink, you can generate C code for use in embedded applications that include fuzzy logic.

3.4 Building a fuzzy inference system:

Fuzzy inference is a method that interprets the values in the input vector and, based on user defined rules, assigns values to the output vector. Using the GUI editors and viewers in the Fuzzy Logic Toolbox, you can build the rules set, define the membership functions, and analyze the behavior of a fuzzy inference system (FIS). The following editors and viewers are provided.

3.5 Key features:

- Specialized GUIs for building fuzzy inference systems and viewing and analyzing results
- Membership functions for creating fuzzy inference systems
- Support for AND, OR, and NOT logic in user-defined rules
- Standard Mamdani and Sugeno-type fuzzy inference systems

- Automated membership function shaping through neuroadaptive and fuzzy clustering learning techniques
- Ability to embed a fuzzy inference system in a Simulink model
- Ability to generate embeddable C code or stand-alone executable fuzzy inference engines.

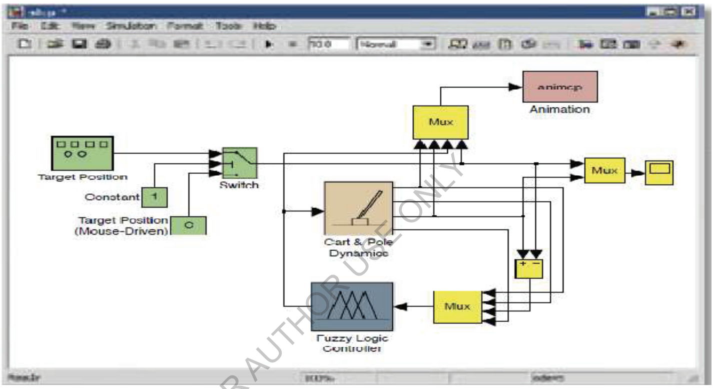


Fig.3.1 Fuzzy interference system

In this section we'll be building a simple tipping example using the graphical user interface (GUI) tools provided by the Fuzzy Logic Toolbox. Although it's possible to use the Fuzzy Logic Toolbox by working strictly from the command line, in general it's much easier to build a system graphically. There are five primary GUI tools for building, editing, and observing fuzzy inference systems in the Fuzzy Logic Toolbox. The Fuzzy Inference System or FIS Editor, the Membership

Function Editor, the Rule Editor, the Rule Viewer, and the Surface Viewer. These GUIs are dynamically linked, in that changes you make to the FIS using one of them, can affect what you see on any of the other open GUIs. You can have any or all of them open for any given system. These are shown in Fig.1

The FIS Editor handles the high level issues for the system: How many input and output variables? What are their names? The Fuzzy Logic Toolbox doesn't limit the number of inputs. However, the number of inputs may be limited by the available memory of your machine. If the number of inputs is too large, or the number of membership functions is too big, then it may also be difficult to analyze the FIS using the other GUI tools.

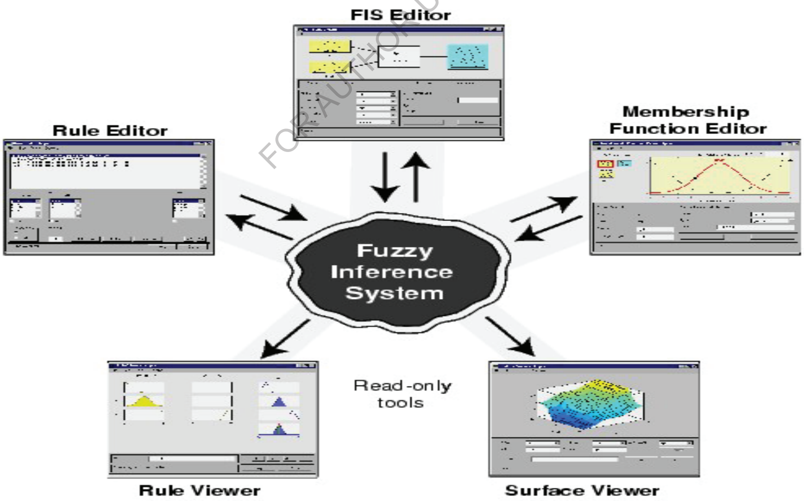


Fig.3.2 The Primary GUI Tools of the Fuzzy Logic Toolbox

The Membership Function Editor is used to define the shapes of all the membership functions associated with each variable. The Rule Editor is for editing the list of rules that defines the behavior of the system.

The Rule Viewer and the Surface Viewer are used for looking at, as opposed to editing, the FIS. They are strictly read-only tools. The Rule Viewer is a matlab-based display of the fuzzy inference diagram shown at the end of the last section. Used as a diagnostic, it can show (for example) which rules are active, or how individual membership function shapes are influencing the results. The Surface Viewer is used to display the dependency of one of the outputs on any one or two of the inputs that is, it generates and plots an output surface map for the system.

The five primary GUIs can all interact and exchange information. Any one of them can read and write both to the workspace and to the disk (the read-only viewers can still exchange plots with the workspace and/or the disk). For any fuzzy inference system, any or all of these five GUIs may be open. If more than one of these editors is open for a single system, the various GUI windows are aware of the existence of the others, and will, if necessary, update related windows. Thus if the names of the membership functions are changed using the Membership Function Editor, those changes are reflected in the rules shown in the Rule Editor. The editors for any number of different FIS systems may be open simultaneously. The FIS Editor, the Membership

Function Editor, and the Rule Editor can all read and modify the FIS data, but the Rule Viewer and the Surface Viewer do not modify the FIS data in any way.

We'll start with a basic description of a two-input, one-output tipping problem. The Basic Tipping Problem. Given a number between 0 and 10 that represents the quality of service at a restaurant (where 10 is excellent), and another number between 0 and 10 that represents the quality of the food at that restaurant (again, 10 is excellent), what should the tip be?

The starting point is to write down the three golden rules of tipping, based on years of personal experience in restaurants.

1. If the service is poor or the food is rancid, then tip is cheap.
2. If the service is good, then tip is average.
3. If the service is excellent or the food is delicious, then tip is generous.

We'll assume that an average tip is 15%, a generous tip is 25%, and a cheap tip is 5%. It's also useful to have a vague idea of what the tipping function should look like. A simple tipping function is shown as in Fig.2. Obviously the numbers and the shape of the curve are subject to local traditions, cultural bias, and so on, but the three rules are pretty universal. Now we know the rules, and we have an idea of what the output should look like. Let's begin working with the GUI tools to construct a fuzzy inference system for this decision process.

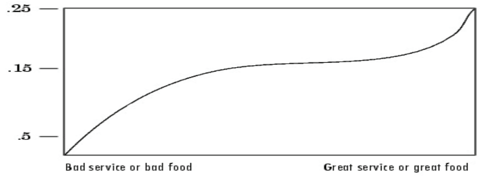


Fig.3.3 The Tipping Function

3.6 The FIS editor:

The following discussion walks you through building a new fuzzy inference system from scratch. If you want to save time and follow along quickly, you can load the already built system by typing fuzzy tipper This will load the FIS associated with the file tipper.fis (the .fis is implied) and launch the FIS Editor. However, if you load the pre-built system, you will not be building rules and constructing membership functions.

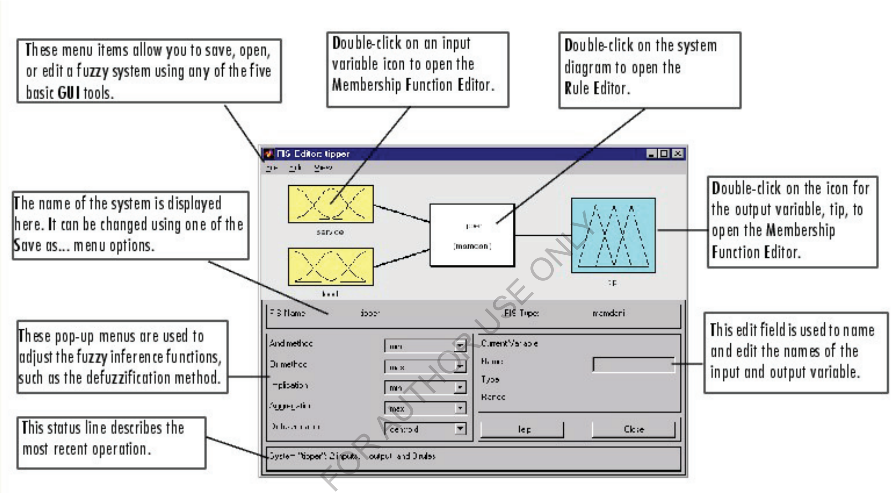
The FIS Editor displays general information about a fuzzy inference system. There's a simple diagram as shown in Fig.3 that shows the names of each input variable on the left, and those of each output variable on the right. The sample membership functions shown in the boxes are just icons and do not depict the actual shapes of the membership functions.

Below the diagram is the name of the system and the type of inference used. The default, Madman-type inference, is what we'll continue to use for this example. Another slightly different type of inference, called Surgeon-type inference, is also available. Below the name of the fuzzy inference system, on the left side of the figure, are the pop-up menus that allow you to modify the various pieces of the inference process. On the right side at the bottom of the figure is the area that displays the name of an input or output variable, its associated membership function type, and its range. The latter two fields are specified only after the membership functions have been. Below that region are the Help and Close buttons that call up online help and close the window, respectively. At the bottom is a status line that relays information about the system.

To start this system from scratch, type fuzzy at the mat lab prompt. The generic untitled FIS Editor opens, with one input, labeled input1, and one output, labeled output1. For this example, we will construct a two-input, one output system, so go to the Edit menu and select Add input. A second yellow box labeled input2 will appear. The two inputs we have in our example are service and food. Our one output is tip.We'd like to change the variable names to reflect that, though:

- Click once on the left-hand (yellow) box marked input1 (the box will be highlighted in red).
- In the white edit field on the right, change input1 to service and press Return.
- Click once on the left-hand (yellow) box marked **i**nput2 (the box will be highlighted in red).
- In the white edit field on the right, change input2 to food and press Return.
- Click once on the right-hand (blue) box marked output1.

- In the white edit field on the right, change output1 to tip.
- From the File menu select Save to workspace as..and a window appears as shown in fig.4.
- Enter the variable name tipper and click on ok.



These menu items allow you to save, open, or edit a fuzzy system using any of the five basic **GUI** tools.

Double-click on an input variable icon to open the Membership Function Editor.

Double-click on the system diagram to open the Rule Editor.

The name of the system is displayed here. It can be changed using one of the Save as... menu options.

Double-click on the icon for the output variable, tip, to open the Membership Function Editor.

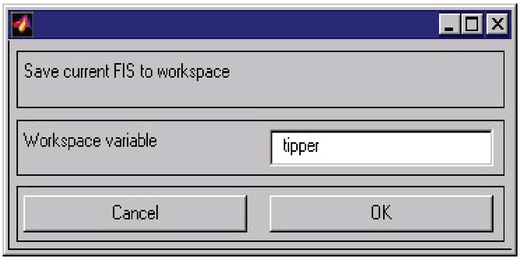
These pop-up menus are used to adjust the fuzzy inference functions, such as the defuzzification method.

This edit field is used to name and edit the names of the input and output variable.

This status line describes the most recent operation.

Fig 3.4 The FIS Editor

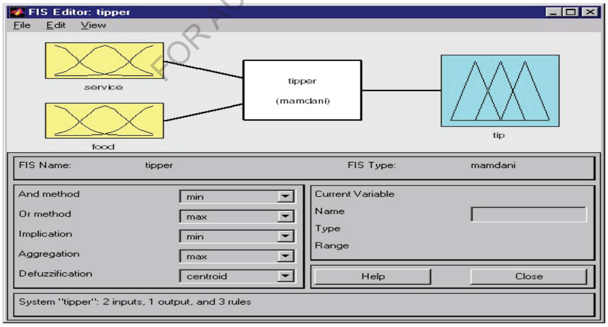
You will see the diagram updated to reflect the new names of the input and output variables. There is now a new variable in the workspace called tipper that contains all the information about this system.


Save current FIS to workspace

Workspace variable: tipper

Fig.3.5‘Save to workspace as...’ Window

By saving to the workspace with a new name, you also rename the entire system. Your window will look like as shown in Fig.5.



FIS Name: tipper

FIS Type: mamdani

And method: min

Or method: max

Implication: min

Aggregation: max

Defuzzification: centroid

Current Variable

Name

Type

Range

System "tipper": 2 inputs, 1 output, and 3 rules

Fig.3.6 The updated FIS Editor

Leave the inference options in the lower left in their default positions for now. You've entered all the information you need for this particular GUI. Next define the membership functions associated with each of the variables. To do this, open the Membership Function Editor. You can open the Membership Function Editor in one of three ways:

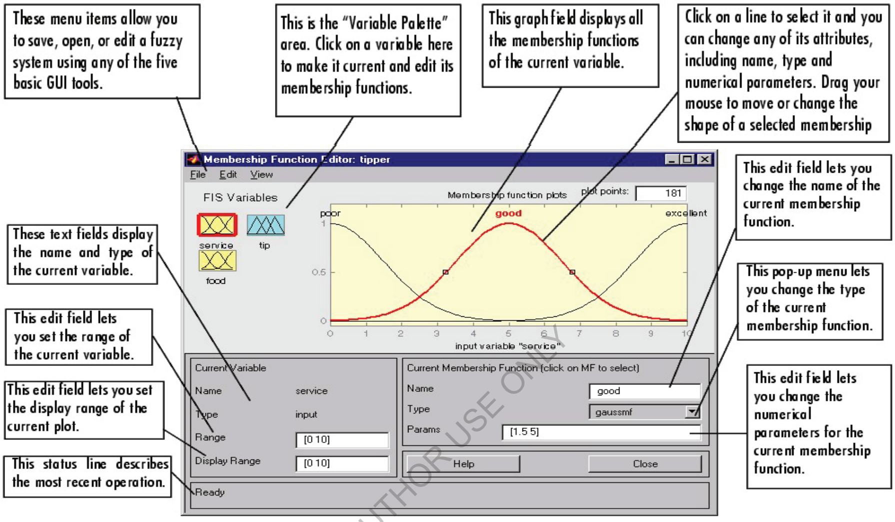
- ✓ Pull down the View menu item and select Edit Membership Functions....
- ✓ Double-click on the icon for the output variable, tip.
- ✓ Type mfedit at the command line.

3.7 The membership function editor:

The Membership Function Editor shares some features with the FIS Editor. In fact, all of the five basic GUI tools have similar menu options, status lines, and Help and Close buttons. The Membership Function Editor is the tool that lets you display and edits all of the membership functions associated with all of the input and output variables for the entire fuzzy inference system. Fig.6 shows the Membership Function Editor.

When you open the Membership Function Editor to work on a fuzzy inference system that does not already exist in the workspace, there is not yet any membership functions associated with the variables that you have just defined with the FIS Editor.

On the upper left side of the graph area in the Membership Function Editor is a "Variable Palette" that lets you set the membership functions for a given variable. To set up your membership functions associated with an input or an output variable for the FIS, select an FIS variable in this region by clicking on it.



plot points: 181

Name: service

Type: input

Range: [0 10]

Display Range: [0 10]

Name: good

Type: gaussmf

Params: [1.5 5]

Ready

These menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

This is the "Variable Palette" area. Click on a variable here to make it current and edit its membership functions.

This graph field displays all the membership functions of the current variable.

Click on a line to select it and you can change any of its attributes, including name, type and numerical parameters. Drag your mouse to move or change the shape of a selected membership

These text fields display the name and type of the current variable.

This edit field lets you set the range of the current variable.

This edit field lets you set the display range of the current plot.

This status line describes the most recent operation.

This edit field lets you change the name of the current membership function.

This pop-up menu lets you change the type of the current membership function.

This edit field lets you change the numerical parameters for the current membership function.

Fig.3.7 The Membership Function Editor

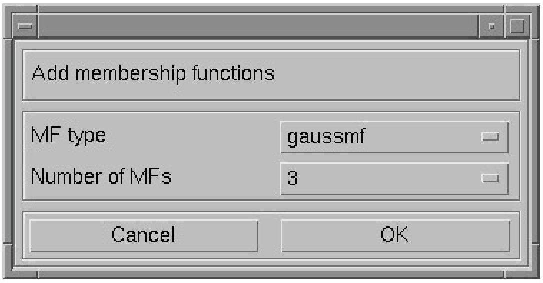
Next select the Edit pull-down menu, and choose Add MFs.... A new window will appear, which allows you to select both the membership function type and the number of membership functions associated with the selected variable. In the lower right corner of the window are the controls that let you change the name, type, and parameters (shape), of the membership function, once it has been selected.

The membership functions from the current variable are displayed in the main graph. These membership functions can be manipulated in two ways. You can first use the mouse to select a particular membership function associated with a given variable quality, (such as poor, for the variable, service), and then drag the membership function from side to side. This will affect the mathematical description of the quality associated with that membership function for a given variable. The selected membership function can also be tagged for dilation or contraction by clicking on the small square drag points on the membership function, and then dragging the function with the mouse toward the outside, for dilation, or toward the inside, for contraction. This will change the parameters associated with that membership function.

Below the Variable Palette is some information about the type and name of the current variable. There is a text field in this region that lets you change the limits of the current variable's range (universe of discourse) and another that lets you set the limits of the current plot (which has no real effect on the system).

The process of specifying the input membership functions for this two input tipper problem is as follows:

- Select the input variable, service, by double-clicking on it. Set both the Range and the Display Range to the vector [0 10].
- Select Add MFs... from the Edit menu. A window pops open as shown in Fig.7.



MF type: gaussmf
Number of MFs: 3

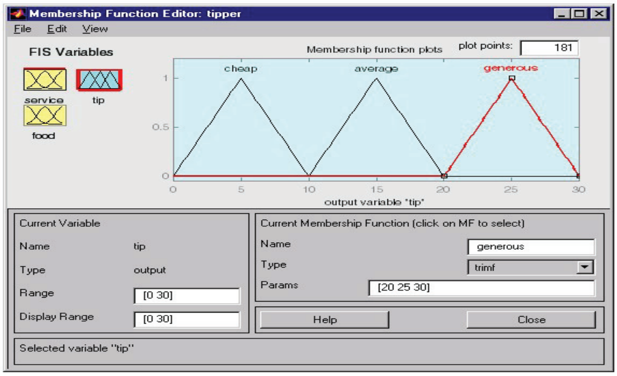
fig.3.8 Add MFs… Window

- Use the pull-down tab to choose gaussmf for MF Type and 3 for Number of MFs. This adds three Gaussian curves to the input variable service.
- Click once on the curve with the leftmost hump. Change the name of the curve to poor. To adjust the shape of the membership function, either use the mouse, as described above, or type in a desired parameter change, and then click on the membership function. The default parameter listing for this curve is [1.5 0].
- Name the curve with the middle hump, good, and the curve with the rightmost hump, excellent. Reset the associated parameters if desired.
- Select the input variable, food, by clicking on it. Set both the Range and the Display Range to the vector [0 10].
- Select Add MFs... from the Edit menu and add two trapmf curves to the input variable food.

- Click once directly on the curve with the leftmost trapezoid. Change the name of the curve to rancid. To adjust the shape of the membership function, either use the mouse, as described above, or type in a desired parameter change, and then click on the membership function. The default parameter listing for this curve is [0 0 1 3].
- Name the curve with the rightmost trapezoid, delicious, and reset the associated parameters if desired.

Next you need to create the membership functions for the output variable, tip. To create the output variable membership functions, use the Variable Palette on the left, selecting the output variable, tip. The inputs ranged from 0 to 10, but the output scale is going to be a tip between 5 and 25 percent.

Use triangular membership function types for the output. First, set the Range (and the Display Range) to [0 30], to cover the output range. Initially, the cheap membership function will have the parameters [0 5 10], the average membership function will be [10 15 20],and the generous membership function will be [20 25 30].Your system should look something like shown in Fig.8.



plot points: 181

Current Variable
Name tip
Type output
Range [0 30]
Display Range [0 30]

Current Membership Function (click on MF to select)
Name generous
Type trimf
Params [20 25 30]

Selected variable "tip"

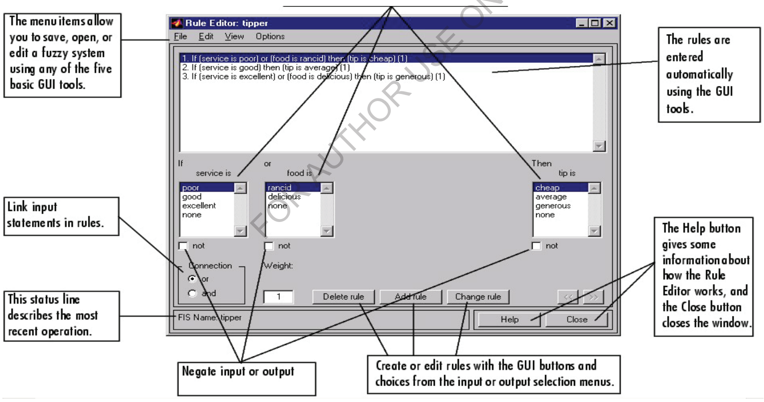
Fig.3.9The updated Membership Function Editor

Now that the variables have been named, and the membership functions have appropriate shapes and names, you're ready to write down the rules. To call up the Rule Editor, go to the View menu and select Edit rules..., or type ruled it at the command line. The Rule Editor window pops open as shown in Fig 6.9

3.8 The rule editor:

Constructing rules using the graphical Rule Editor interface is fairly self-evident. Based on the descriptions of the input and output variables defined with the FIS Editor, the Rule Editor allows you to construct the rule statements automatically, by clicking on and selecting one item in each input variable box, one item in each output box, and one connection item. Choosing none as one of the variable qualities will exclude that variable from a given rule. Choosing not under any variable name will negate the associated quality. Rules may be changed, deleted, or added, by clicking on the appropriate button.

The Rule Editor also has some familiar landmarks, similar to those in the FIS Editor and the Membership Function Editor, including the menu bar and the status line. The Format pop-up menu is available from the Options pull-down menu from the top menu bar -- this is used to set the format for the display. Similarly, Language can be set from under Options as well. The Help button will bring up a MATLAB Help window.



1. If (service is poor) or (food is rancid) then (tip is cheap) (1)
2. If (service is good) then (tip is average) (1)
3. If (service is excellent) or (food is delicious) then (tip is generous) (1)

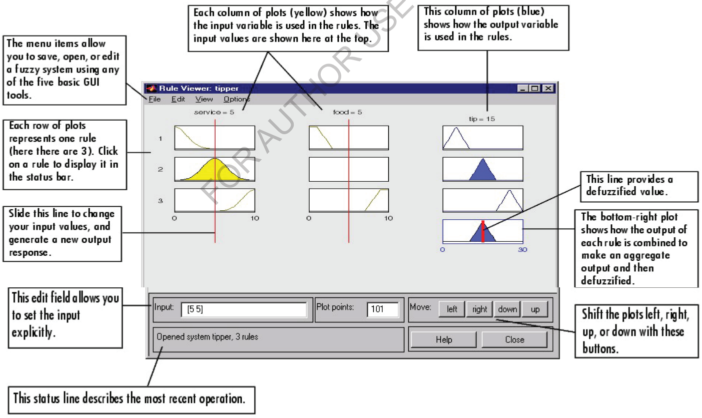
If service is: poor
or food is: rancid
Then tip is: cheap
Connection: ● or ○ and
Weight: 1
FIS Name: tipper

Fig.3.10The Rule Editor

To insert the first rule in the Rule Editor, select the following:

3.9 The rule viewer:

The Rule Viewer displays a roadmap of the whole fuzzy inference process. It's based on the fuzzy inference diagram described in the previous section. You see a single figure window as shown in fig.10 with 10 small plots nested in it. The three small plots across the top of the figure represent the antecedent and consequent of the first rule. Each rule is a row of plots, and each column is a variable. The first two columns of plots (the six yellow plots) show the membership functions referenced by the antecedent, or the if-part of each rule. The third column of plots (the three blue plots) shows the membership functions referenced by the consequent, or the then-part of each rule.



Input: [5 5]

Plot points: 101

Opened system tipper, 3 rules

Each column of plots (yellow) shows how the input variable is used in the rules. The input values are shown here at the top.

This column of plots (blue) shows how the output variable is used in the rules.

The menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

Each row of plots represents one rule (here there are 3). Click on a rule to display it in the status bar.

Slide this line to change your input values, and generate a new output response.

This edit field allows you to set the input explicitly.

This status line describes the most recent operation.

This line provides a defuzzified value.

The bottom-right plot shows how the output of each rule is combined to make an aggregate output and then defuzzified.

Shift the plots left, right, up, or down with these buttons.

Fig.3.11 The Rule Viewer

If you click once on a rule number, the corresponding rule will be displayed at the bottom of the figure. Notice that under food, there is a plot which is blank. This corresponds to the characterization of none for the variable food in the second rule. The fourth plot in the third column of plots represents the aggregate weighted decision for the given inference system. This decision will depend on the input values for the system.

There are also the now familiar items like the status line and the menu bar. In the lower right there is a text field into which you can enter specific input values. For the two-input system, you will enter an input vector, [9 8], for example, and then click on input. You can also adjust these input values by clicking anywhere on any of the three plots for each input. This will move the red index line horizontally, to the point where you have clicked. You can also just click and drag this line in order to change the input values. When you release the line, (or after manually specifying the input), a new calculation is performed, and you can see the whole fuzzy inference process take place. Where the index line representing service crosses the membership function line "service is poor" in the upper left plot will determine the degree to which rule one is activated.

A yellow patch of color under the actual membership function curve is used to make the fuzzy membership value visually apparent. Each of the characterizations of each of the variables is specified with respect to the input index line in this manner. If we follow rule 1 across the top of the diagram, we can see the consequent "tip is cheap" has been truncated to exactly the same degree as the (composite) antecedent--this is the implication process in action. The aggregation occurs down the third column, and the resultant aggregate plot is shown in the single plot to be found in the lower right corner of the plot field. The de-fuzzyfied output value is shown by the thick line passing through the aggregate fuzzy set.

The Rule Viewer allows you to interpret the entire fuzzy inference process at once. The Rule Viewer also shows how the shape of certain membership functions influences the overall result. Since it plots every part of every rule, it can become unwieldy for particularly large systems, but, for a relatively small number of inputs and outputs, it performs well (depending on how much screen space you devote to it) with up to 30 rules and as many as 6 or 7 variables.

The Rule Viewer shows one calculation at a time and in great detail. In this sense, it presents a sort of micro view of the fuzzy inference system. If you want to see the entire output surface of your system, that is, the entire span of the output set based on the entire span of the input set, you need to open up the Surface Viewer. This is the last of our five basic GUI tools in the Fuzzy Logic Toolbox, and you open it by selecting View surface... from the Viewmenu. The Surface Viewer window pops open as shown in fig.10

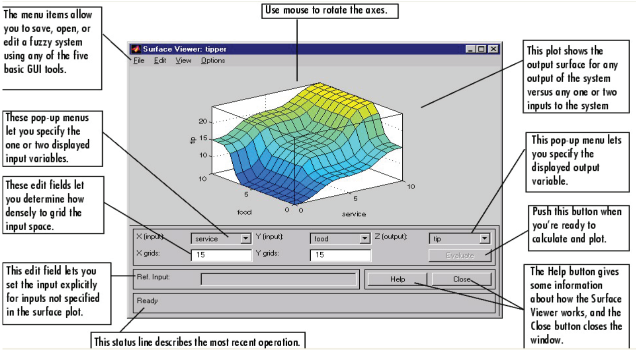
3.10 The surface viewer:

Upon opening the Surface Viewer, we are presented with a two-dimensional curve that represents the mapping from service quality to tip amount. Since this is a one-input one-output case, we can see the entire mapping in one plot. Two-input one-output systems also work well, as they generate three-dimensional plots that mat lab can adeptly manage. When we move beyond three dimensions overall, we start to encounter trouble displaying the results. Accordingly, the

Surface Viewer is equipped with pop-up menus that let you select any two inputs and any one output for plotting.

Just below the pop-up menus are two text input fields that let you determine how many x-axis and y-axis grid lines you want to include. This allows you to keep the calculation time reasonable for complex problems. Pushing the Evaluate button initiates the calculation, and the plot comes up soon after the calculation is complete. To change the x-axis or y-axis grid after the surface is in view, simply change the appropriate text field, and click on either X-grids or Y-grids, according to which text field you changed, to redraw the plot.The Surface Viewer has a special capability that is very helpful in cases with two (or more) inputs and one output: you can actually grab the axes and reposition them to get a different three-dimensional view on the data.

The Ref. Input field is used in situations when there are more inputs required by the system than the surface is mapping. Suppose you have a four-input one-output system and would like to see the output surface. The Surface Viewer can generate a three-dimensional output surface where any two of the inputs vary, but two of the inputs must be held constant since computer monitors cannot display a five-dimensional shape. In such a case the input would be a four-dimensional vector with Na Ns holding the place of the varying inputs while numerical values would indicate those values that remain fixed. An Na N is the IEEE symbol for "not a number."



The menu items allow you to save, open, or edit a fuzzy system using any of the five basic GUI tools.

Use mouse to rotate the axes.

This plot shows the output surface for any output of the system versus any one or two inputs to the system

These pop-up menus let you specify the one or two displayed input variables.

This pop-up menu lets you specify the displayed output variable.

These edit fields let you determine how densely to grid the input space.

Push this button when you're ready to calculate and plot.

This edit field lets you set the input explicitly for inputs not specified in the surface plot.

The Help button gives some information about how the Surface Viewer works, and the Close button closes the window.

This status line describes the most recent operation.

Fig.3.12 The Surface Viewer

This concludes the quick walk-through of each of the main GUI tools. Notice that for the tipping problem, the output of the fuzzy system matches our original idea of the shape of the fuzzy mapping from service to tip fairly well. In hindsight, you might say, "Why bother? I could have just drawn a quick lookup table and been done an hour ago!" However, if you are interested in solving an entire class of similar decision-making problems, fuzzy logic may provide an appropriate tool for the solution, given its ease with which a system can be quickly modified.

3.11 Importing and exporting from the gui tools:

When you save a fuzzy system to disk, you're saving an ascii text FIS file representation of that system with the file suffix .fis. This text file can be edited and modified and is simple to understand. When you save your fuzzy system to the mat lab workspace, you're creating a variable (whose name you choose) that will act as a mat lab structure for the FIS system. FIS files and FIS structures represent the same system.

In FLC, basic control action is determined by a set of linguistic rules. These rules are determined by the system. Since the numerical variables are converted into linguistic variables, mathematical modeling of the system is not required in FC. The FLC comprises of three parts: fuzzification, interference engine and defuzzification.

The FC is characterized as i. seven fuzzy sets for each input and output. ii. Triangular membership functions for simplicity. iii. Fuzzification using continuous universe of discourse. iv. Implication using Mamdani's, 'min' operator. v. Defuzzification using the height method.

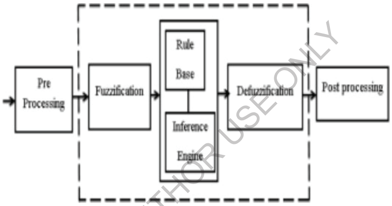


Fig.3.13 Fuzzy logic controller

3.12 Fuzzification

Membership function values are assigned to the linguistic variables, using seven fuzzy subsets: NB (Negative Big), NM (Negative Medium), NS (Negative Small), ZE (Zero), PS (Positive Small), PM (Positive Medium), and PB (Positive Big). The partition of fuzzy subsets and the shape of membership CE(k) E(k) function adapt the shape up to appropriate system. The value of input error and change in error are normalized by an input scaling factor

Table: Fuzzy Rules

$e/\Delta e$	LP	MP	SP	S	SN	MN	LN
LP	PB	PB	PB	PM	PM	PS	Z
MP	PB	PB	PM	PM	PS	Z	NS
SP	PB	PM	PM	PS	Z	NS	NM
S	PM	PM	PS	Z	NS	NM	NM
SN	PM	PS	Z	NS	NM	NM	NB
MN	PS	Z	NS	NM	NM	NB	NB
LN	Z	NS	NM	NM	NB	NB	NB

In this system the input scaling factor has been designed such that input values are between -1 and +1. The triangular shape of the membership function of this arrangement presumes that for any particular E(k) input there is only one dominant fuzzy subset. The input error for the FLC is given as

$$E(k) = \frac{P_{ph(k)} - P_{ph(k-1)}}{V_{ph(k)} - V_{ph(k-1)}} \qquad (14)$$

$$CE(k) = E(k) - E(k-1) \qquad (15)$$

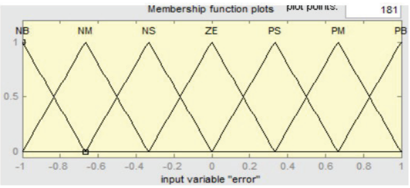


Fig.3.14Membership functions

Inference Method: Several composition methods such as Max–Min and Max-Dot have been proposed in the literature. In this paper Min method is used. The output membership function of each rule is given by the minimum operator and maximum operator. Table 1 shows rule base of the FLC.

Defuzzification: As a plant usually requires a non-fuzzy value of control, a defuzzification stage is needed. To compute the output of the FLC, „height" method is used and the FLC output modifies the control output. Further, the output of FLC controls the switch in the inverter. In UPQC, the active power, reactive power, terminal voltage of the line and capacitor voltage are required to be maintained. In order to control these parameters, they are sensed and compared with the reference values. To achieve this, the membership functions of FC are: error, change in error and output.

3.13 CONTROL TECHNIQUES

1. The role of the soft starting mode is to generate the reference value of the supply current in a manner that limits the starting current of the IM at a preset value. The actual current of IM (Im) is measured and its RMS value is evaluated by RMS detector. The command or preset value of the

motor current (Im∗) and its actual value (Im) are compared. The comparison resulted error is passed into a proportional integral (PI) controller to generate the command motor current (Is∗). Limiting the starting current provides a smooth acceleration and reduces the torque pulsations of IM during soft starting period.

2. **SPEED CONTROL MODE** There are several methods for controlling the speed of three-phase IMs. These methods can be classified into two main categories according to the control side of the IM: a) speed control methods through the stator such as changing the applied frequency, changing the applied voltage, changing the number of the stator poles and voltage/frequency (v/f) control, and b) speed control methods through the rotor such as rotor resistance control and rotor slip power recovery. Variable frequency drives (VFDs) are the commercial drives. Speed control by VFDs is based on changing both the stator voltage and frequency of the IM. VFDs are widely used for wide-range variable-speed IM applications. However, they are very expensive and hence not convenient when they are used for limited-range variable-speed IM applications. Since the proposed speed control strategy depends on changing the stator voltage only, so it is simple, low cost and more convenient for limited-range variable-speed IM application which is intended in this research. The role of the speed control mode is to generate the reference current value (Is∗) in a way that makes the measured speed of IM (ωm) follows the command speed (ωm∗). Command

and measured speed are compared and the difference is used as an input signal to a PI speed controller to generate $Is*$.

3. **PFC CONTROL** Since PWM AC/AC choppers can only modify the magnitude of the applied voltage, they are normally negatively viewed; when they are used in IM drive systems, for their low PF. Therefore, the main contribution of the proposed control strategy is achieving high PF approximately unity as in case of resistive loads. The proposed PFC strategy was implemented during starting and speed control operating modes of IM drive while using AC chopper. The principle of harmonic minimization of the proposed control strategy depends on using PWM technique. Whereas, the principle of reactive power management is to obtain PFC depends on

the proposed current control technique; in which the actual supply currents are forced to track their reference currents that are in phase with supply voltages. The role of PFC block is to continuously correct the input PF during IM operation. Fig. 4a shows the proposed PFC using HBCC technique. The reference value of the stator current ($Is*$) is utilized to obtain the three phase reference supply currents ($i*sa$, $i*sb$ and $i*sc$) by multiplying the value ($Is*$) by unit vectors of the supply voltages (usa, usb and usc) as:

$$\begin{bmatrix} i_{sa}^* \\ i_{sb}^* \\ i_{sc}^* \end{bmatrix} = I_s^* \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u_{sa} \\ u_{sb} \\ u_{sc} \end{bmatrix}$$

The unit vectors (usa, usb and usc) are generated by measuring the phase voltages of the supply (vsa, vsb and vsc) and by using three zero crossing detection (ZCD) circuits and three lookup tables as shown in Fig. 4a. The reference currents of the supply (i∗sa, i∗sb and i∗sc), are compared with their corresponding actual values (isa, isb and isc) respectively. The resulted errors are passed through three hysteresis bands (HBs) and their outputs are the three switching signals (S1, S2 and S3). Operation of HBCC technique to obtain the switching signal (S1) is explained by Fig. 4b. The logic control signals block is utilized to find the higher value of the supply phase voltages. The switching signal F is generated from the three switching signals (S1, S2 and S3) and the three logic control signals (q1, q2 and q3) as:

$$F = S1 \times q1 + S2 \times q2 + S3 \times q3$$

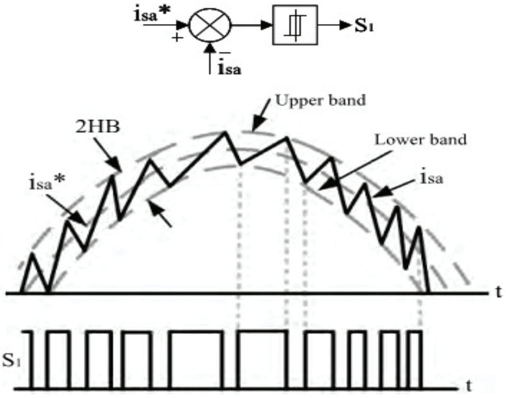


Fig.3.15 Generation of Pulse S_1.

3.14 HYSTERIS BAND CURRENT CONTROL

The current control strategy plays an important role in the development of shunt active filter. The hysteresis-band current control method (Anshumanshukla et al 2007) is popularly used because of its simplicity in implementation. Hysteresis current controller derives the switching signals of the inverter power switches in a manner that reduces the current error. The switches are controlled asynchronously to ramp the current through the inductor up and down so that it follows the reference.

When the current through the inductor exceeds the upper hysteresis limit, a negative voltage is applied by the inverter to the inductor. This causes the current through the inductor to decrease. Once the current reaches the lower hysteresis limit, a positive voltage is applied by the inverter through the inductor and this causes the current to increase and the cycle repeats. The current controllers of the three phases are designed to operate independently. They determine the switching signals to the respective phase of the inverter This method has the drawbacks of variable switching frequency, heavy interference between the phases in case of three phase active filter with isolated neutral and irregularity of the modulation pulse position (Simone et al 2000). These drawbacks result in high current ripples, acoustic noise and difficulty in designing input filter. In this chapter, a constant frequency hysteresis current controller is proposed for shunt active filter applications. The details of the proposed current control strategy are presented in the next section.

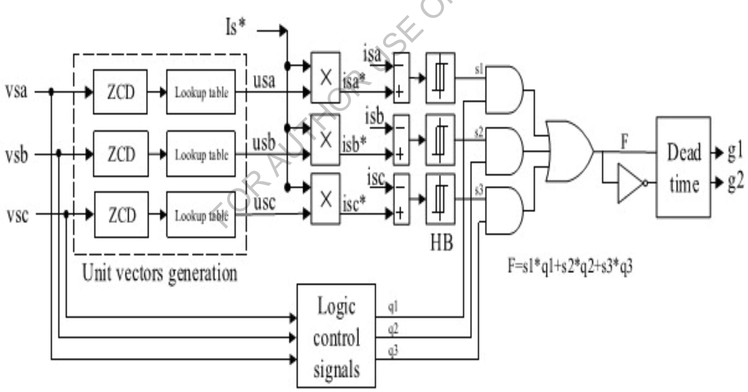


Fig.3.16 Proposed HBCC technique

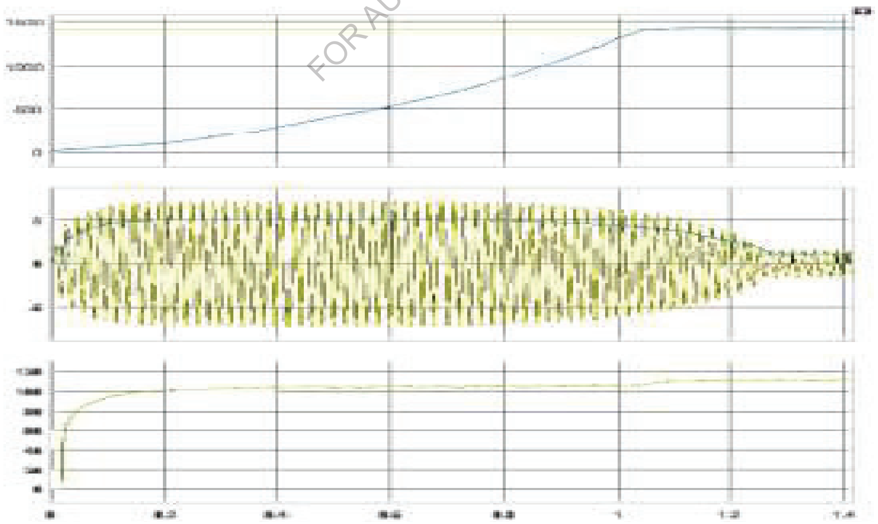
CHAPTER – 4

RESULTS AND DISCUSSION

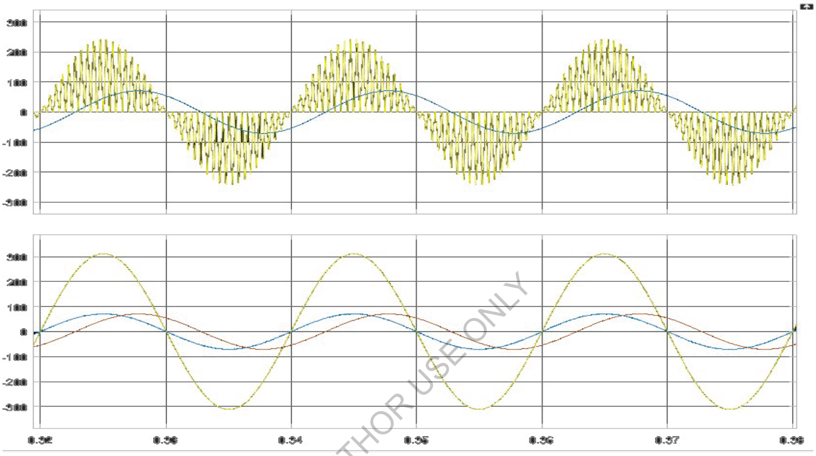
4.1 INTRODUCTION

The proposed AC chopper is simulated in the MATLAB/ Simulink environment and a prototype model is implemented. The simulation was used to confirm the proposed control strategy theoretically. While the experimental prototype has been constructed to confirm the proposed strategy experimentally. Four test cases are examined. Corresponding simulation and experimental results are obtained and compared. Below table shows the parameters used inthis paper.

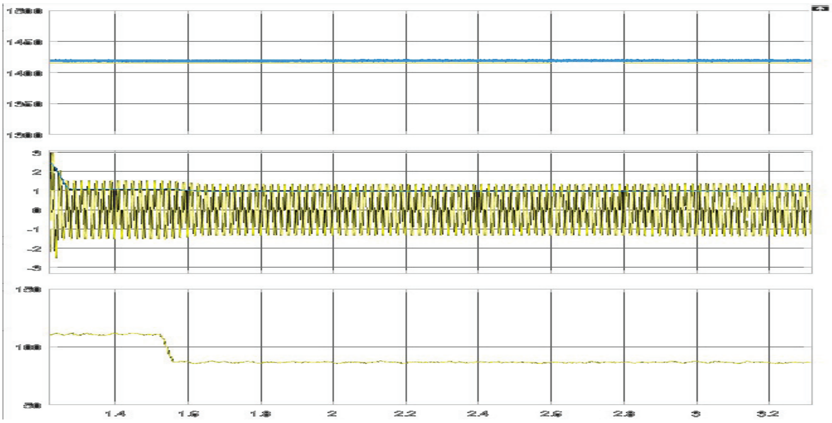
4.2 MATHEMATICAL ANALYSIS



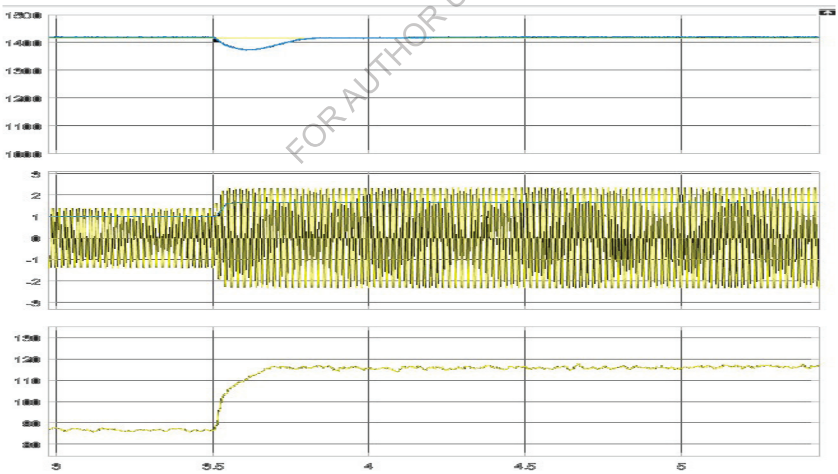
4.1 Starting of the motor with Proposed PWM AC Chopper



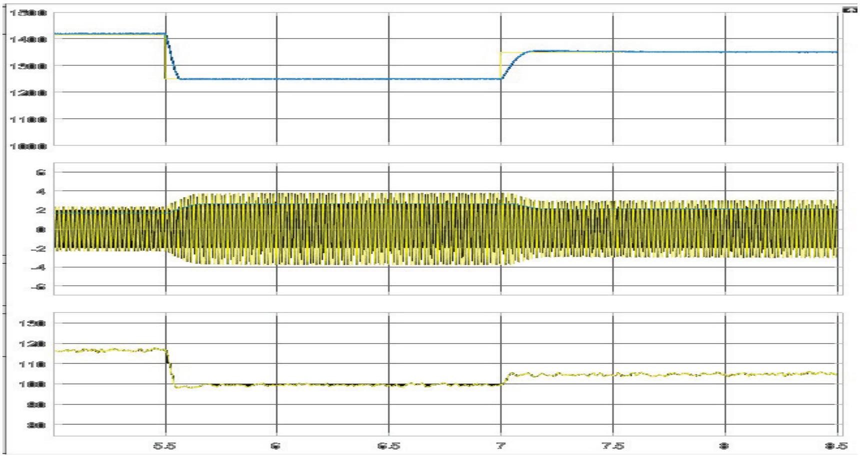
4.2 PFC of the drive system during start up of the IM



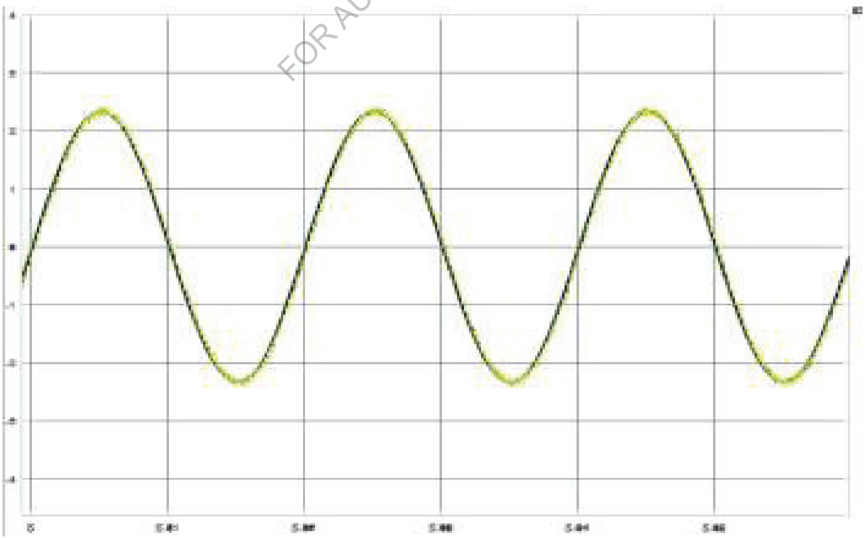
4.3 Variation of the motor speed, the current and phase voltage at activation of the speed controller



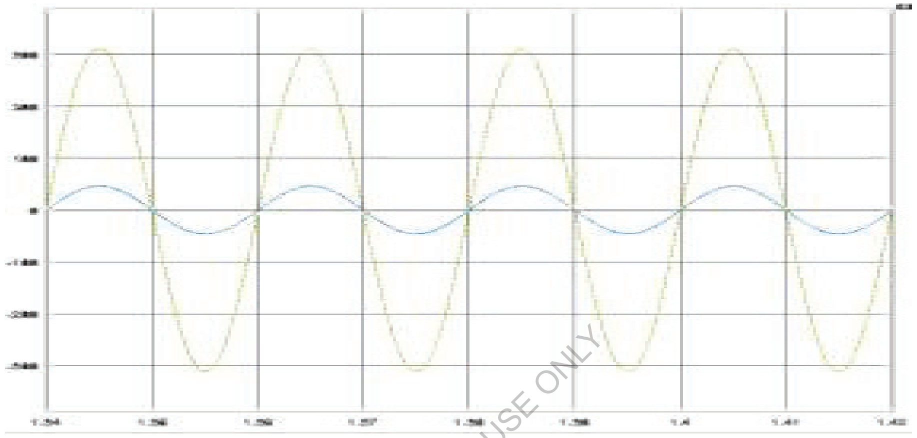
4.4 Variation of the motor Speed, Current and Phase Voltage at step change in the load torque



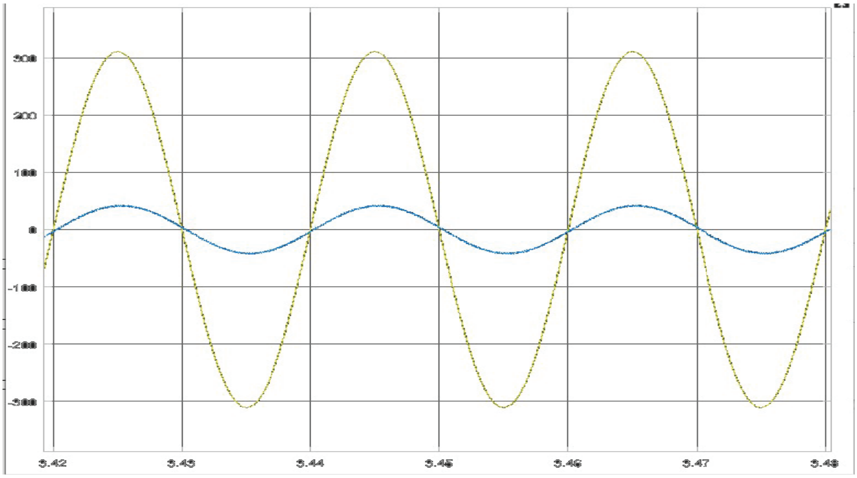
4.5 Variation of the motor Speed, Current and Phase Voltage at step change in the reference speed



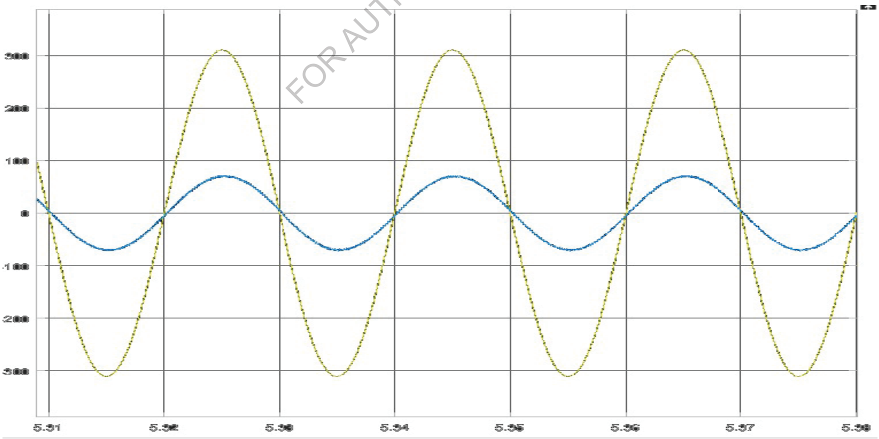
4.6 Reference and measured currents supply



4.7 Supply voltage and current at different testing cases of the proposed system



4.8Supply voltage and current at different testing cases of the proposed system



4.9 Supply voltage and current at different testing cases of the proposed system

CHAPTER – 5

CONCLUSION AND FUTURE ENHANCEMENT

5.1 CONCLUSION

A new control strategy of three-phase squirrel cage IM fed from PWM AC chopper has been simulated and laboratory implemented using d SPACE (DS1104) control board. The main control objective is to correct the input PF with different operating conditions of the induction motor drive system. Input PFC is achieved by forcing the actual currents of the chopper to track their reference currents that are in phase with the input voltages using HBCC technique. The proposed control strategy uses only two PWM signals for driving the active switches of the AC chopper. The proposed system is simple, reliable and low cost as it has only four IGBT switches. Operation principle and mathematical analysis of the proposed system are introduced. The system was simulated using MATLAB/SIMULINK and a laboratory system was implemented. The effectiveness of the proposed control strategy has been tested at starting, reference speed change and load torque variation. The obtained results from the experimental and computer simulation works verify the validity of the proposed control strategy during all testing conditions.Performance of the system without PFC is roughly compared in accordance with concerning the proposed PFC technique during the three test cases. Comparative results illustrate that the system with the proposed PFC technique has a corrected PF and hence a better performance.

REFERENCES

[1] T. Mishima, Y. Nakagawa, and M. Nakaoka, ‘‘A bridgeless BHB ZVS-PWM AC–AC converter for high-frequency induction heating applications,’’ *IEEE Trans. Ind. Appl*, vol. 51, no. 4, pp. 3304–3315, Jul./Aug. 2015. doi: 10.1109/TIA.2015.2409177.

[2] S. Mahendran, I. Gnanambal, and A. Maheswari, ‘‘FPGA-based genetic algorithm implementation for AC chopper fed induction motor,’’ *Int. J. Electron*, vol. 103, no. 12, pp. 2029–2041, Apr. 2016. doi: 10.1080/ 00207217.2016.1175034.

Fuzzy Logic Controller Based Three-Phase PWM AC Chopper Fed Induction Motor Drive with HBCC Technique

https://iaeme.com/Home/journal/IJEET 236 editor@iaeme.com

[3] F. Luo, H. Ye, and M. H. Rashid, *Digital Power Electronics and Applications*. Amsterdam, The Netherlands: Elsevier, 2005.

[4] L. Rajaji, C. Kumar, and M. Vasudevan, ‘‘Fuzzy and ANFIS based soft starter fed induction motor drive for high performance applications,’’ *ARPN J. Eng. Appl. Sci.*, vol. 3, no. 4, pp. 12–24, Aug. 2008.

[5] A. Gastli and M. M. Ahmed, ‘‘ANN-based soft starting of voltage controlled-fed IM drive system,’’ *IEEE Trans. Energy Convers*, vol. 20, no. 3, pp. 497–503, Sep. 2005.doi: 10.1109/TEC.2004.841522.

[6] M. Muchlas and H. Soetedjo, ‘‘Use of the maximum torque sensor to reduce the starting current in the induction motor,’’ *Sensors Transducers*, vol. 114, no. 3, pp. 161–169, Mar. 2010.

[7] G. Zenginobuz, I. Cadirci, M. Ermis, and C. Barlak, ‘‘Performance optimization of induction motors during voltage-controlled soft starting,’’ *IEEE Trans. Energy Convers.*, vol. 19, no. 2, pp. 278–288, Jun. 2004. doi: 10.1109/TEC.2003.822292.

[8] K. Sundareswaran and P. S. Nayak, ‘‘Ant colony-based feedback controller design for soft-starter fed induction motor drive,’’ *Appl. Soft Comput.* , vol. 12, no. 5, pp. 1566–1573, May 2012. doi: 10.1016/j.asoc.2011.12.012.

[9] K. Sundareswaran, N. Rajasekar, and V. T. Sreedevi, ‘‘Performance comparison of capacitor-run induction motors supplied from AC voltage regulator and SPWM AC chopper,’’ *IEEE Trans. Ind. Electron.* , vol. 53, no. 3, pp. 990–993, Jun. 2006. doi: 10.1109/TIE.2006.874256.

[10] S. Jothibasu and M. K. Mishra, ‘‘An improved direct AC–AC converter for voltage sag mitigation,’’ *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 21–29, Jan. 2015. doi: 10.1109/TIE.2014.2334668.

[11] Z. Chen, C. Mao, D. Wang, J. Lu, and Y. Zhou, ‘‘Design and implementation of voltage source converter excitation system to improve power system stability,’’ *IEEE Trans. Ind. Appl.*, vol. 52, no. 4, pp. 2778–2788, Jul./Aug. 2016. doi: 10.1109/TIA.2016.2543685.

[12] D. Yildirim and M. Bilgic, ‘‘PWM AC chopper control of single-phase induction motor for variable-speed fan application,’’ in *Proc. IEEE IECON*, Orlando, FL, USA, Nov. 2008, pp. 1337–1342.

[13] J. M. Flores-Arias, A. M. MuÃśoz, F. D. Perez, V. Pallares-Lopez, and D. Gutierrez, ‘‘Voltage regulator system based on a PWM AC chopper converter,’’ in *Proc. IEEE ISIE*, Gdansk, Poland, Jun. 2011, pp. 468–473.

[14] K. Sundareswaran and A. P. Kumar, ‘‘Voltage harmonic elimination in PWM AC chopper using genetic algorithm,’’ *IEE Proc. - Electric Power Appl.*, vol. 151, no. 1, pp. 26–31, Jan. 2004. doi: 10.1049/ip-epa:20040061.

[15] K. Georgakas and A. Safacas, ‘‘Modified sinusoidal pulse-width modulation operation technique of an AC–AC single-phase converter to optimise the power factor,’’ *IET Power Electron.*, vol. 3, no. 3, pp. 454–464, May 2010. doi: 10.1049/iet-pel.2009.0092.

[16] D.-H. Jang and G.-H.Choe, ‘‘Improvement of input power factor in AC choppers using asymmetrical PWM technique,’’ *IEEE Trans. Ind. Electron.*, vol. 42, no. 2, pp. 179–185, Apr. 1995. doi: 10.1109/41.370384.

[17] P. Sanjit and A. Aurasopon, ‘‘Asymmetrical PWM for harmonics reduction and power factor improvement in PWM AC choppers using bee colony optimization,’’ *J. Power Electron.*, vol. 15, no. 1, pp. 227–234, Jan. 2015. doi: 10.6113/JPE.2015.15.1.227.