Particle Swarm Optimization based Adaptive Neuro-Fuzzy Inference System for MPPT Control of a Three Phase Grid Connected Photovoltaic System

by

Jeffrey Andrew-Cotter

Submitted in partial fulfillment of the requirements for the

degree of Master of Science in Electrical and Computer Engineering

Lakehead University

Thunder Bay, Ontario

July 2019

Abstract

There is a significant demand for renewable energy systems to ensure sustainable and environmentally friendly living. Shifting from the use of fossil fuels to renewable energy sources will decrease both the reliance on these fossil fuels and associated pollution. By decreasing greenhouse gas emissions from energy generation, human induced global warming and environment destruction will slow. According to Environment and Climate Change Canada, Canada has committed, along with other leading countries in greenhouse gas emissions, to maintain the total global temperature increase below 2°C. Solar photovoltaic (PV) energy systems are of particular interest due to the availability and portability of such systems. According to Natural Resources Canada (NRCan), Canada's reliance on solar energy as a renewable energy source is rapidly growing. NRCan claims that Canada's total quantity of installed solar energy reached 1834 MW in 2014.

Therefore, the main objective of this thesis is to design an intelligent controller-based efficient solar energy conversion system in order to meet the growing demand for clean energy. Solar energy systems consist of a PV cell array (solar panel) that uses the sunlight to generate direct current (DC) power. To maximize efficiency of the energy conversion, a novel maximum power point tracking (MPPT) algorithm is developed to deliver maximum power from the PV panel to the load. The conversion system must be designed to transfer maximum power regardless of the intensity of the sunlight and size of the load. Due to this requirement, the buck boost converter with an intelligent controller generating its control signal is the ideal solution. The converter is able to both step up and step down the input hence transferring maximum possible power at all times. Intelligent algorithms do not need exact mathematical models of the system and can handle any nonlinearity of the system. As an intelligent controller, a neuro-fuzzy controller (NFC), specifically an adaptive neuro-fuzzy inference system (ANFIS), will be developed to generate the control signal for the DC-DC converter while coping with variable weather conditions. A hybrid training algorithm is developed that implements particle swarm optimization to train nonlinear system parameters and the least squares estimator to train the linear parameters. The power at the output of the DC-DC converter can be either stored directly in batteries or converted to alternating current (AC) power. For simulation purposes of this thesis, the DC power available at the output of the converter is fed into a three phase, two level

voltage source inverter that is controlled using proportional-integral controllers to control the d and q axis output voltages. Three phase output with constant amplitude and constant frequency is required to connect the system to the grid. The AC inverter output is filtered with an L filter and is interfaced with the grid to achieve effective grid connection.

Simulations of the proposed energy conversion system and the proposed ANFIS training algorithm are completed in MATLAB/Simulink. The simulation results prove the effectiveness of the designed ANFIS and proposed training algorithm as the ANFIS-based MPPT controller is able to extract maximum power from the solar panel for varying irradiance conditions. The simulations further prove that grid connection is possible while obtaining three phase output voltage and current with low total harmonic distortion. The real-time implementation of the system is performed using the dSPACE DS1104 development board for communication to and from Simulink running on a PC. The proposed ANFIS-based MPPT controller and the proposed training algorithm are verified in real-time for a wide range of irradiance condition and changes in load. As determined by real-time implementation, however, the grid connection poses a significant challenge due to unknown factors in the Centennial building at Lakehead University as well as a lack of funds preventing the purchase of vital equipment. As such, stand-alone mode of operation is attempted in which the output of the buck boost converter is connected to a resistive load. The real time results prove the efficacy of the proposed ANFIS-based control and training algorithms.

Acknowledgements

I would first like to thank Dr. M. N. Uddin for his support throughout my undergraduate and Master degree at Lakehead University. He provided useful information regarding publishing papers and finalizing this thesis.

I would also like to thank Mr. Ifte Amin. Throughout my entire Master degree, Ifte would always be more than glad to lend a hand even with his own work to complete. Further, Ifte's superior knowledge provided very useful for constructive conversations relating to this thesis. Ifte's support is immensely appreciated.

Another thank you to Dr. Natarajan for always being available to ask questions and provide support. Dr. Natarajan assisted immensely in troubleshooting issues that arose during the real time implementation stage of this thesis. Without his aid and his vast practical and theoretical knowledge, the real time implementation would have been much more problematic.

Thank you to the professors and students who attended the thesis seminar and thesis defense for their time, asking questions and providing useful feedback. Special thanks are due to Mr. Bruce Misner, Mr. Warren Paju and Mr. Dan Vasiliu who provided various electrical components, useful input and new ideas related to my research.

Lastly, I would like to thank my good friend Nicholas Frayn who was also working on his Master's thesis concurrently to me under the supervision of Dr. Uddin. Thank you Nick for the valued conversations and the constant support you provided to me.

Table of Contents

List of Figures

List of Tables

List of Symbols

Symbol	Meaning
I_g	Solar cell photon generated current
I_s	Reverse saturation current of the diode
I	Solar cell output current
n	Ideality factor
V_T	Thermal voltage
R_s	Solar cell series resistance
R_p	Solar cell shunt resistance
D	Duty cycle of the converter
T_s	Period
i_L	Inductor current
i_s	Source current
i_D	Diode current
t_{on}	Switch ‘on’ time
V_L	Inductor voltage
Δi_L	Inductor current ripple
V_o	Output voltage
L	Inductance value
c	Capacitance value
P_{in}	Input power
P_{out}	Output power
I_o	Output current
I_1	Minimum inductor current
I_2	Maximum inductor current
Δv_o	Output voltage ripple
I_{OB}	Boundary current
V_{An}, V_{Bn}, V_{Cn}	Line to neutral (phase) voltages
V_{AB}, V_{BC}, V_{CA}	Line to line (line) voltages

V_{dc}	Inverter dc input voltage
p, q, r	Linear parameters in the consequent part of the fuzzy system
$W_{1,2,3,4}$	Firing strengths for each rule
$f_{1,2,3,4}$	Consequent part output functions
$\mu_{A,B}$	Input membership functions
Θ_{k+1}	Linear parameter estimate
N	Number of training data pairs
V_{ini}	Initial particle velocity
V_{mod}	New particle velocity
p_{best}	Particle best fitness value
g_{best}	Globally best fitness value among all particles
v	Particle velocity
w	Inertia weight
C_1	Personal learning rate
C_2	Global learning rate
x	Particle's current position
R_L	Load resistance
R_s	Source resistance
V_{mpp}	PV voltage at the MPP
I_{mpp}	PV current at the MPP
R_{opt}	Optimal resistance to achieve operation at the MPP
V_d, V_q	DC signals in the rotating reference frame
θ	The angle between phase 'a' in the time varying reference frame and the q axis in the rotating reference frame
V_a, V_b, V_c	The three phase signals in the time varying abc reference frame
m_a	Amplitude modulation index
m_f	Frequency modulation index
V_{ref}	dq reference voltage
f_c	LC filter cut-off frequency
f_s	Switching frequency

List of Abbreviations

Abbreviation	Meaning
NRC	Natural Resources Canada
PV	Photovoltaic
DC	Direct current
MPPT	Maximum power point tracking
NFC	Neuro-fuzzy controller
AC	Alternating current
P&O	Perturb and observe
IncC	Incremental conductance
FL	Fuzzy logic
ANFIS	Adaptive neuro-fuzzy inference system
PSO	Particle swarm optimization
SEPIC	Single-ended primary inductor converter
THD	Total harmonic distortion
PI	Proportional-integral
ANN	Artificial neural network
dq	Direct-quadrature
MPP	Maximum power point
CCM	Continuous conduction mode
VSI	Voltage source inverter
PWM	Pulse width modulation
TSK-1	Tagaki-Sugeno-Kang-1
PID	Proportional-Integral-Derivative
RLSE	Recursive least squares estimator
LSE	Least-squares estimator
GD	Gradient descent
PLL	Phase locked loop
RMS	Root mean square

KVL	Kirchhoff's voltage law
CCM	Continuous conduction mode
RMSE	Root mean square error
FFT	Fast Fourier Transform

Chapter 1

Introduction

1.1 Research Motivation

As a result of the shift away from fossil fuel energy sources, advances in renewable energy systems have increased the efficiency and decreased the cost of such systems. Of particular interest are solar photovoltaic systems as they are readily available and portable. However, depending on the load connected to the solar array, the maximum power might not be extracted from the solar array if no control is implemented. It is desirable to control the solar array in order that maximum power is always extracted regardless of weather or load conditions. This type of control is known as maximum power point tracking (MPPT) and can be performed in a variety of ways [3, 6-15, 19-31, 33-37]. Currently in industry, the most common MPPT controller is known as Perturb and Observe. This method has a myriad of problems which motivates the requirement for more effective MPPT controllers to improve system efficiency by extracting maximum power from the array with little to no oscillations once the maximum power point is reached. Usually, a DC-DC power electronic converter [16-18] is implemented to match the load impedance to the solar array's output impedance which forces the solar array to output its maximum power for a given set of weather conditions. The power extracted from the solar array can either be used to charge a battery or converted into AC for either single phase operation or grid connection [32]. As such, the main objective of this thesis is to develop an efficient solar energy conversion system that implements an intelligent MPPT controller [1, 2, 4, 5]. The proposed system is an improvement to the existing MPPT methods and is a step towards the necessity for clean renewable energy systems as a replacement to fossil fuel-based energy generation.

1.2 Literature Review

This section of the thesis provides an overview of the existing maximum power point tracking control techniques for DC-DC converters. Significant research has been completed based on conventional MPPT techniques such as Perturb and Observe (P&O) and Incremental Conductance (IncC) [7, 9-13]. Substantial work has also been completed that focuses on intelligent fuzzy logic (FL) based MPPT [7, 11, 13]. Some work has been completed on other intelligent controllers such as neural networks and adaptive neuro-fuzzy inference system (ANFIS) based control algorithms [3, 5-8, 14]. Maximum power point tracking methods can be classified into two categories:

1. Conventional MPPT controllers
2. Intelligent MPPT controllers

1.2.1 Conventional MPPT Controllers

Conventional maximum power point tracking algorithms/controllers are relatively simple to implement but inherent oscillations occur at the maximum power point. This results in a reduction of the available power that can be transferred to the DC link. To improve the efficiency of the conventional methods, additional control efforts or modified techniques are required. Most research now is based on intelligent controllers due to their inherent benefits and uses the conventional techniques as a comparison to validate the new intelligent controller based research.

In [10] the authors have provided a detailed analysis with simulation and experimental results of various MPPT methods for PV systems. Specifically, they find that both incremental conductance and perturb and observe have very similar responses with a similar tracking efficiency. Many other conventional MPPT methods are also compared in this work. Similarly, [11] provides another comparison of MPPT methods focussing on P&O, fuzzy logic and backstepping control. The authors found that their proposed backstepping controller outperformed both fuzzy logic and P&O based tracking algorithms. In [12] a hybrid MPPT

algorithm is proposed that combines P&O and the fractional short circuit current measurement methods. It is proven that the proposed hybrid algorithm is significantly more effective at tracking the MPP with reduced oscillations. The authors prove by simulation and experimental results that their proposed hybrid algorithm has less power loss from oscillations and a high convergence speed at the downfall of the more complex nature of the proposed system. Generally, improving the efficiency of the MPPT algorithm comes at a cost of a higher complexity and a longer computation time. In [13] the authors provide a wide variety of available MPPT techniques and analyses them based on various factors such as cost, complexity, applications, type of converter etc. This work only outlines each method with corresponding theory and provides no design, simulation or implementation of the discussed methods. [7] provides a theoretical discussion of both conventional and intelligent MPPT algorithms. This work provides advantages and disadvantages of the algorithms and allows for quick reference of various conventional MPPT features. [9] proposes an adaptive P&O technique in which a PI controller is used to calculate the perturb value used in the P&O algorithm and applies it to a microgrid. In this case, DC-AC conversion is not required as the authors store the DC energy directly in a battery for storage purposes in a microgrid.

1.2.2 Intelligent MPPT Controllers

Intelligent maximum power point tracking controllers are generally more complex and more difficult to implement as compared to the conventional ones. These methods however, provide a much better response with increased efficiency and better tracking of the maximum power point. Intelligent controllers are classified into three categories: fuzzy logic, neural networks and ANFIS. Fuzzy logic controllers generally require an expert knowledge of the system under consideration while neural networks rely on training data and training algorithms to tune link weights that model the desired system. ANFIS is a hybrid system that combines features of both fuzzy logic and neural networks. The main benefits of intelligent controllers are: they operate without any system model, they handle any nonlinearity in the system well and they do not depend on system parameters. These three benefits make intelligent controllers of growing interest for MPPT implementation in PV solar energy conversion systems.

Many works have analyzed fuzzy logic based MPPT, neural network based MPPT and ANFIS based MPPT. In all previous research the intelligent methods outperform the conventional MPPT methods. In [3], the authors compare a variety of intelligent MPPT algorithms and discuss their associated features. They assert that neuro-fuzzy systems are medium to complex algorithms, have a fast tracking speed with high efficiency and are a medium to complex implementation. They develop an ANFIS and use MATLAB's built in NeuroFuzzyDesigner to create and train the ANFIS. The authors of [5] propose an ANFIS that uses a modified particle swarm optimization (PSO) to train the system in which they implement the selection operator from genetic algorithms to remove the worst member of the population during training with PSO. This work only discusses ANFIS training and does not apply the ANFIS to a solar energy conversion system for maximum power point tracking. In [6], an improved PSO is used directly to search for the maximum power point. The authors do not use the PSO for training an ANFIS but instead use it for online MPP searching. This method eliminates almost all oscillations once the maximum power point is reached. [8] proposes a fuzzy logic controller for maximum power point tracking and applies the controller to a single-ended primary inductor converter (SEPIC) converter. The author further applies a fuzzy logic controller to a single phase inverter in order to generate a single phase AC voltage. This previous research is the closest to the full practical grid connected system that is discussed in this thesis. The authors of this work prove that the fuzzy controller is able to effectively track the maximum power point with a higher power extraction than a conventional proportional-integral (PI) controller based MPPT. They further prove that the designed inverter control is effective at creating the desired sinusoidal output voltage and current with a total harmonic distortion (THD) less than 5%. The authors of [14] design a basic ANFIS using the NeuroFuzzyDesigner and apply the ANFIS directly to a quasi-z-source inverter for maximum power point tracking and DC-three phase AC conversion. Grid connection was not completed in this work as the inverter was not connected to the grid. Applying MPPT directly to the inverter removes the need for a DC-DC converter used for impedance matching and hence maximum power point tracking. As a result, the system in [14] is less expensive to build than the conventional systems that use a DC-DC converter for maximum power point tracking and an inverter for DC-AC conversion. Further, [14] does not provide any comparisons between the proposed algorithms and conventional or alternate MPPT methods.

Since intelligent methods are more complicated and some require parameter training, faster computations are required in order to maintain the same operating speed as the simpler conventional methods. As a result, intelligent MPPT controller implementation is more financially costly than the implementation of conventional methods. These aforementioned works have mainly focussed only on the DC-DC converter and have not converted the DC signals into three phase AC signals through the use of an inverter for grid connection. It is of interest to consider DC-AC conversion for grid connection thus creating a practical energy conversion system. Therefore, this thesis develops a grid connected PV solar energy conversion system as a more practical system than the previous work in this area.

1.3 Objectives

The main objective of this thesis is to design, test and implement an adaptive neuro-fuzzy inference system based maximum power point tracking algorithm to control a buck boost converter. Further extension is also completed applying an inverter to generate a three phase AC signal for grid connection. In practical implementation, the goal is to prove the operation of the proposed training algorithm and the ANFIS based MPPT controller. This thesis will satisfy the following objectives:

a. To design a buck boost converter to be used as an impedance matcher that will guarantee maximum power delivery to the dc link
b. To design an adaptive neuro-fuzzy inference system based maximum power point tracking algorithm and to design the control scheme for the inverter
c. To develop a recursive least squares estimator – particle swarm optimization based training algorithm for ANFIS parameter training
d. To simulate the proposed system and validate its operation by comparing it to a conventional P&O based system
e. To perform real time implementation of the proposed system to confirm the operation of the proposed ANFIS based MPPT algorithm

1.4 Thesis Organization

Chapters in this thesis focus on main topics and the subsections in each chapter focus on a specific detail or feature relating to the main topic. Chapter 1 provides the motivation for the work outlined in this thesis and includes a review of previous works relating to maximum power point tracking algorithms. Chapter 2 provides detailed derivations of the system model: solar panel model, buck boost converter, and 3 phase voltage source inverter. Chapter 3 discusses the theory relating to intelligent controllers such as fuzzy logic, artificial neural networks (ANN) and ANFIS. Chapter 4 discusses maximum power transfer, the perturb and observe MPPT algorithm, theory pertaining to three phase voltage source inverters: control schemes, direct-quadrature (dq) reference frame transformations, and filter theory. Chapter 5 outlines the design of the buck boost converter and the proposed MPPT algorithm. Chapter 6 provides the results from MATLAB/Simulink simulations. The performance of the proposed algorithm is validated by altering weather conditions and comparing the response to the conventional P&O response. Chapter 7 delivers the results from real time implementation of the proposed MPPT controller. Chapter 8 offers a summary of the thesis and results obtained from simulation and real time implementation and outlines possible future work with the proposed system.

Chapter 2

System Modelling

This chapter provides a discussion on the modelling of solar panels and buck boost converter. Mathematical analysis is provided with associated derivations.

2.1 Solar Photovoltaic Panel Modelling

This section outlines the modelling of a PV cell and the effects of changing irradiance and temperature of the panel. The equivalent circuit of a solar cell is shown in Fig. 2-1. A solar panel is composed many solar cells and a solar array is composed of a combination of solar panels connected in series or in parallel to achieve the desired output voltage and current from the array.

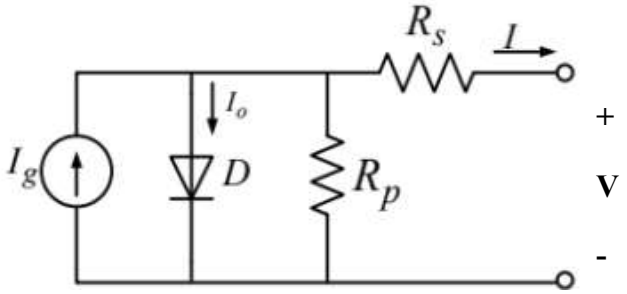

Fig. 2-1: Equivalent circuit of a solar cell.

From the equivalent circuit and Kirchhoff's current law, an equation for the cell's output current can be found as:

$$I = I_g - I_o - I_p \tag{1}$$

$$I = I_g - I_s \left\{ e^{\left[\frac{V+IR_S}{nV_T}\right]} \right\} - \frac{V+IR_S}{R_p} \tag{2}$$

where,

I_g is the generated current from photons

I_s is the reverse saturation current of the diode

I is the output current of the cell

n is the ideality factor

V_T is the thermal voltage

R_s is the series resistance

R_p is the shunt resistance

From knowledge of solar panel operation and construction, it is clear that the current outputted by the panel is directly related to the intensity of the sunlight reaching the panel. Further, the voltage produced by the panel is directly related to the temperature of the panel. If the irradiance increases, the current produced by the panel increases and if the temperature of the panel decreases, the voltage produced by the panel increases. As such, solar panels are more efficient at lower temperatures as they produce a larger voltage for a given irradiance level. It is important to realize the solar panel has a highly nonlinear output impedance which changes significantly with changing weather conditions. Thus, it is difficult to match the output impedance of the panel to a load. If the load is constant, there is only one scenario when the output impedance of the panel is the same as the load and hence maximum power is transferred to the load. As a result, it is desirable to have a controller that will always match the solar panel output impedance to the load impedance. This controller is known as a maximum power point tracker. Maximum power point tracking and associated types of trackers will be discussed in later chapters.

The characteristic equations that model a solar panel are used to plot the relationship between voltage, current and power output of a solar panel. Fig. 2-2 shows the power-voltage and current-voltage curves for a solar panel with varying irradiance and a constant temperature of 25°C. Fig. 2-3 shows the power-voltage and current-voltage curves for a solar panel with varying temperature and a constant irradiance of 1000W/m^2. Note that the LG300N1C-G3 module is used in Simulink for simulation purposes.

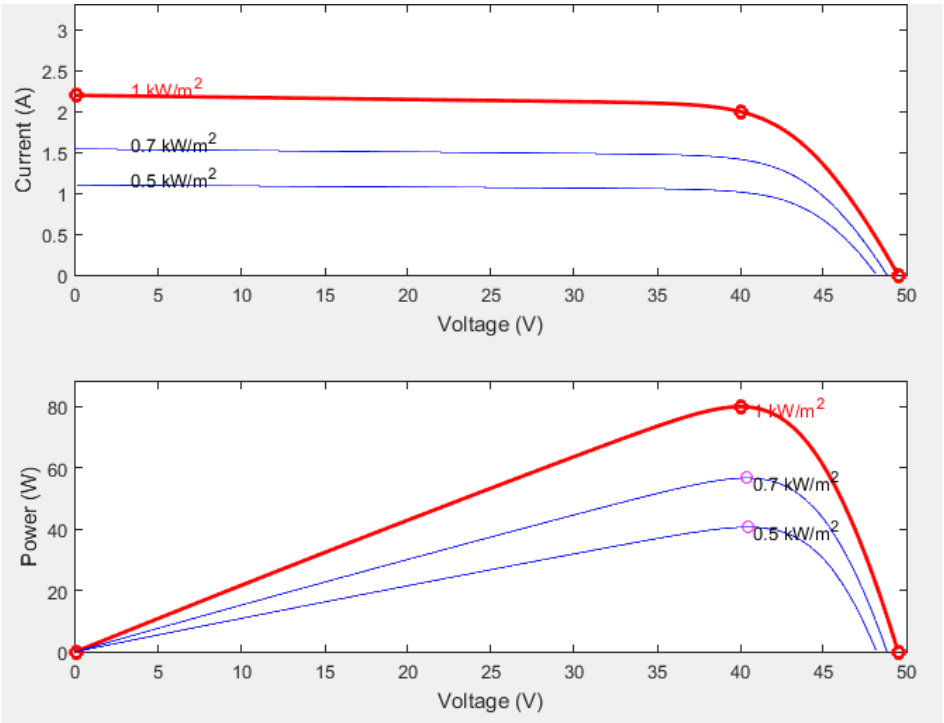

Fig. 2-2: Solar panel characteristic curves with constant temperature and varying irradiance.

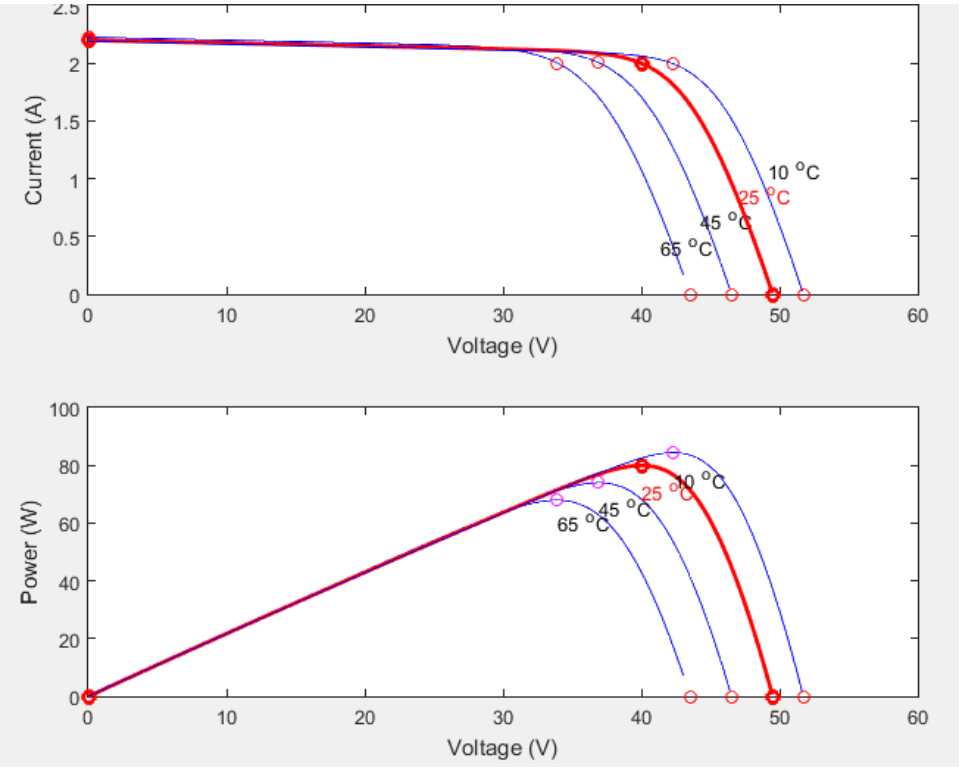

Fig. 2-3: Solar panel characteristic curves with constant irradiance and varying temperature.

It is clearly illustrated in Fig. 2-2 that as the irradiance increases (with a constant temperature), the maximum power point (MPP) current and hence, the power generated by the solar panel increases, while the MPP voltage remains constant. The maximum power points are labelled on the above figures with a circle (please ignore the circle on the horizontal axis). The MPPs occur at the maximum power on the P-V curves. As a higher intensity of sunlight reaches the photodiodes that create the solar panel, more electron-hole pairs are created and hence a larger current is established. It is clearly illustrated in Fig. 2-3 that as the temperature increases (with a constant irradiance) the MPP voltage and hence the power generated by the solar panel decreases while the MPP current remains constant. Any changes in irradiance levels prove to affect the maximum power point significantly more than changes in panel temperature. Therefore, for this thesis, the solar panel is assumed to remain at a constant temperature of 25°C while the irradiance level changes. Further, the solar emulator used for real time implementation has no options for temperature change. The emulator only allows for the irradiance to be varied. Due to these two reasons, the system is designed to operate at a single temperature.

2.2 Buck Boost Converter Modelling

The buck boost converter is a popular DC-DC converter as it has the ability to both step up and step down the input voltage applied to the converter [32], [33]. The converter consists of a switch, a diode, a capacitor and an inductor arranged in a circuit configuration corresponding to the buck boost converter topology. This section provides a theoretical analysis of the inverting buck boost converter. Provided in this section is: the circuit diagram, various waveforms of currents and voltages in the circuit and derivations of various parameters and component values for operation in continuous conduction mode. For the purposes of this thesis, the converter operating in continuous conduction mode (CCM) is considered as it is undesirable to allow the inductor current to reach zero and become discontinuous.

Fig. 2-4 shows the circuit diagram for the inverting buck boost converter. Two modes of operation occur. Fig. 2-5 shows the first mode of operation when the switch is on. Fig. 2-6 shows the second mode of operation when the switch is off. Fig. 2-7 shows the waveforms of various voltages and currents in the converter.

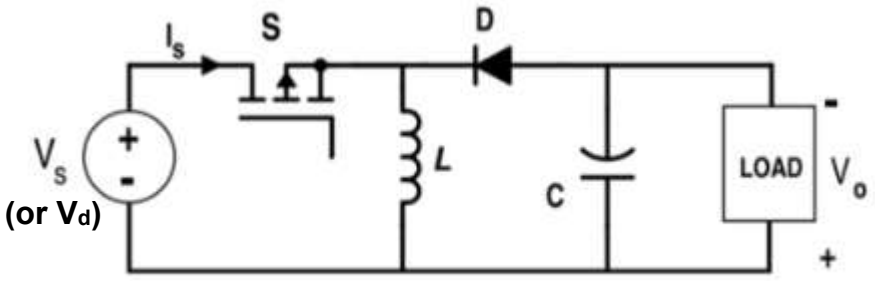

Fig. 2-4: Inverting buck boost converter circuit diagram.

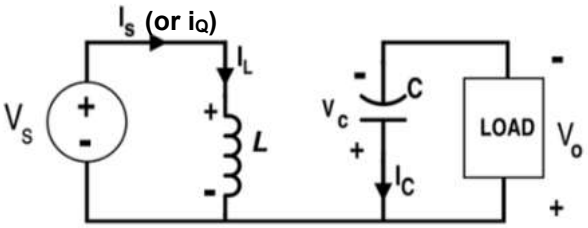

Fig. 2-5: Mode 1 of operation.

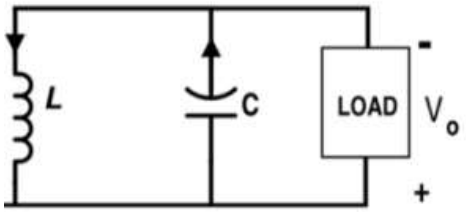

Fig. 2-6: Mode 2 of operation.

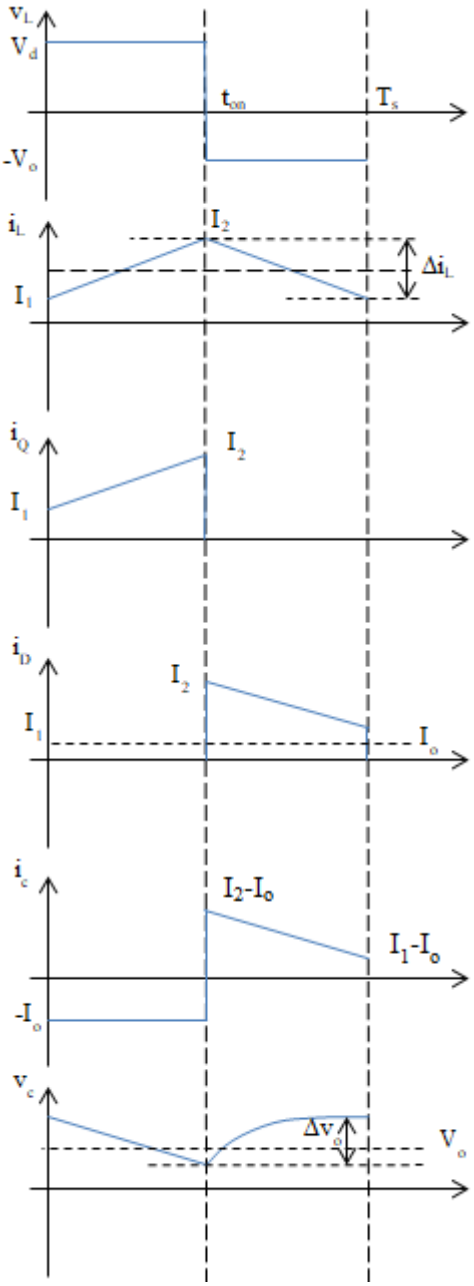

Fig. 2-7: Waveforms of voltages and currents in the converter.

It is important to note that all derivations are done under the assumption that the converter is 100% efficient. Further assume the switch is on for a time of t_{on} (or DT_s).

Using Kirchhoff's current law, it is found that,

$$i_L = i_S + i_D \tag{3}$$

where, D is the duty ratio of the switch

T_s is the period

i_L is the inductor current

i_s is the source current

i_D is the diode current

During mode 1 ($0<t<t_{on}$),

$$v_L = V_s = L\frac{\Delta i_L}{DT_s}$$

$$\Delta i_L = \frac{V_s D T_s}{L} \tag{4}$$

In this mode, the switch is on and the diode is off. Here, the inductor is in series with the source and the inductor's current increases while accumulating energy. The capacitor is in series with the load and hence the capacitor discharges supplying energy to the load.

During mode 2 ($t_{on}<t<T_s$),

$$v_L = -V_o = L\frac{(-\Delta i_L)}{(1-D)T_s}$$

$$\Delta i_L = \frac{V_o(1-D)T_s}{L} \tag{5}$$

In this mode, the switch is on and the diode is off. Here the inductor and capacitor are in parallel with the load. The energy stored in the inductor charges the capacitor and provides energy to the load.

From (4) and (5),

$$\frac{V_o(1-D)T_s}{L} = \frac{V_s D T_s}{L}$$

$$V_o(1-D) = V_s D$$

$$\frac{V_o}{V_s} = \frac{D}{1-D} \tag{6}$$

Another method to derive (6) is as follows:

The average inductor current must be equal to zero.

Therefore,

$$V_s D T_s = V_o (1 - D) T_s$$
$$\frac{V_o}{V_s} = \frac{D}{1-D} = A_v \quad (7)$$

If the duty ratio is greater than 0.5, the output voltage will be larger than the input voltage and hence boost operation is achieved. If the duty ratio is less than or equal to 0.5, the output voltage is less than the input voltage and buck operation is achieved. Therefore, depending on the value of the duty ratio, the buck boost converter is able to both step up and step down input voltages.

Various relations between the average currents of the components can be found. Referring to the converter waveforms it is easily determined that,

$$I_D = I_o = (1 - D) I_L$$
$$I_L = \frac{I_o}{1-D} \quad (8)$$

And,

$$I_s = D I_L \quad (9)$$

where, I_D is the average diode current

I_o is the average output current

I_L is the average inductor current

I_s is the average switch current

Since the converter is assumed to be 100% efficient,

$$P_{in} = P_{out} \quad (10)$$
$$V_s I_{in} = V_o I_o$$

$$\frac{V_o}{V_s} = \frac{I_{in}}{I_o}$$

$$\frac{I_{in}}{I_o} = \frac{D}{1-D} = \frac{1}{A_i} \tag{11}$$

To prove that the buck boost converter can match the load resistance to the optimal PV panel resistance corresponding to the MPP, a relationship between R_L and R_{opt} can be found. From the gains of the converter,

$$V_o = A_v V_{mpp}, \text{ where } V_{mpp}=V_s$$

$$I_o = A_i I_{mpp}, \text{ where } I_{mpp}=I_{in}$$

And,

$$R_{opt} = \frac{V_{mpp}}{I_{mpp}} \tag{12}$$

Then,

$$R_{opt} = \frac{(1-D)^2}{D^2} R_L \tag{13}$$

Therefore, if the duty cycle is greater than 50%, the optimal resistance is less than the load resistance and if the duty cycle is less than 50%, the optimal resistance is greater than the load resistance. As a result, the buck boost converter can match the load to the optimal panel resistance for all weather conditions. The buck converter and boost converter cannot achieve such results. For the buck converter, the optimal resistance is always greater than the load resistance and for the boost converter the optimal resistance is always less than the load resistance. If the load resistance is less than the PV panel's output resistance, then the converter operates in buck mode and if the load resistance is greater than the PV panel's output resistance then the converter operates in boost mode.

To determine the output voltage ripple it is assumed that $I_l \geq I_o$ and the ripple current only flows through the capacitor.

$$q = Cv \tag{14}$$

$$I = \frac{q}{t}$$

$$q = It \tag{15}$$

$$\Delta v_o = \frac{\Delta Q}{c} = \frac{I_o D T_S}{c} \tag{16}$$

Thus, the equations for output ripple voltage and inductor ripple current are:

$$\Delta i_L = \frac{V_S D T_S}{L} \tag{17}$$

$$\Delta v_o = \frac{I_o D T_S}{c} \tag{18}$$

From (17) and (18), the values of the inductor and capacitor can be found as:

$$L = \frac{V_S D T_S}{\Delta i_L} \tag{19}$$

$$c = \frac{I_o D T_S}{\Delta v_o} \tag{20}$$

The converter could operate in discontinuous mode. The boundary between continuous and discontinuous modes of operation is when I_l=0 (See Fig. 2-7).

When I_l=0,

$$I_{LB} = \frac{\Delta i_L}{2} \tag{21}$$

$$I_{LB} = \frac{(1-D) V_o T_S}{2L} \tag{22}$$

$$I_{OB} = (1-D) I_{LB} = \frac{(1-D)^2 V_o T_S}{2L} \tag{23}$$

To ensure operation in continuous mode, $I_o > I_{OB}$. Note that the inductor is chosen so that continuous mode of operation is achieved.

Chapter 3

Artificial Intelligence Theory

This chapter provides a theoretical analysis and explanation of various artificial intelligence systems such as fuzzy logic (FL), artificial neural networks (ANN) and adaptive neuro-fuzzy inference systems (ANFIS). The most commonly used hybrid training method for ANFIS is discussed and a new hybrid algorithm is proposed for ANFIS training.

3.1 Fuzzy Logic

This section outlines the first order Tagaki-Sugeno-Kang (TSK-1) or Sugeno fuzzy reasoning. Note that there are many types of fuzzy reasoning but for this thesis, Sugeno fuzzy reasoning is the focus.

Consider a simple fuzzy system with two inputs (x, y), one output (z), two rules and two membership functions per input. Let $A_1 = \mu_{A1}$ and $A_2 = \mu_{A2}$ be the membership functions for the first input and let $B_1 = \mu_{B1}$ and $B_2 = \mu_{B2}$ be the membership functions for the second input. Further assume the input is fuzzified using the singleton fuzzifier. This means that instead of the input variable being a fuzzy set (or having a membership function), the input is a single value corresponding to the inputted value. For example, if the input is x=2 then the fuzzified input will also be x=2.

The input membership functions can be continuous or piecewise continuous functions. Some commonly used membership functions are: triangular, trapezoidal, Gaussian, Bell or generalized Bell. The designer determines which membership function is best for a given system. To design a fuzzy logic system, no mathematical model of the system is required. This feature is beneficial because many physical systems are complex and difficult to model. Further, fuzzy logic is superior at handling system nonlinearities. If the fuzzy logic system is used as a controller, it is more advantageous than the classical linear controllers such as the proportional-integral-derivative (PID) controller The PID controller cannot effectively control highly nonlinear

systems since a linear approximation around the operating point is usually performed to design a PID controller for a nonlinear system. Also, classical controllers have the downfall of being designed to control a system at a specific operating point. If the operating point is changes, the controller needs to be re-tuned or a different more complex PID controller is required. For a fuzzy logic controller, there is a range of operating points defined by the universe of discourse for a specific application. Therefore, if the operating point changes, the fuzzy logic controller will still be effective at controlling the system under the condition that the new operating point is within the designed universe of discourse.

The fuzzy system is designed as:

R_1: If x is A_1 and y is B_1 then z is f_1

R_2: If x is A_2 and y is B_2 then z is f_2

where, $f_1 = p_1x+q_1y+r_1$

$f_2 = p_2x+q_2y+r_2$

(p, q, r) are linear parameters in the consequent part that will be optimized when used in an ANFIS.

The graphical method for Sugeno fuzzy reasoning is shown below:

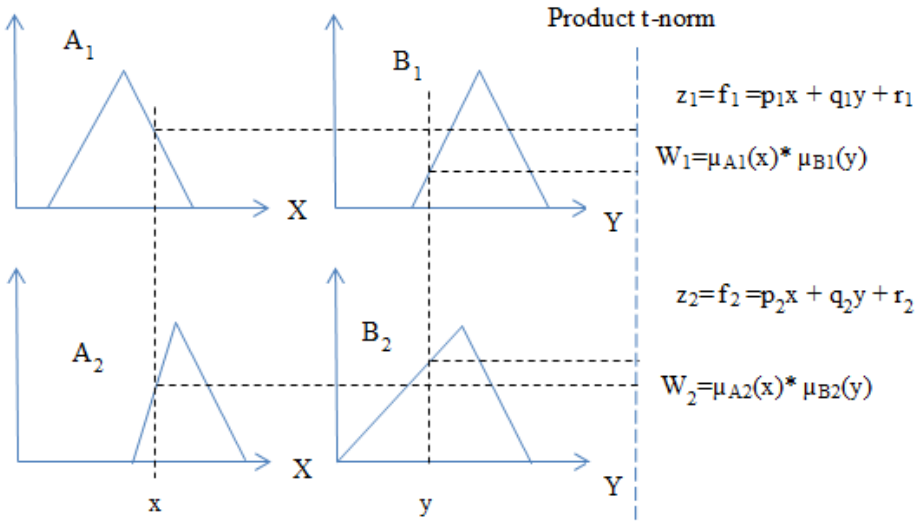

Fig. 3-1: Graphical method for Sugeno fuzzy reasoning.

The defuzzified normalized output of the fuzzy logic system is:

$$z = \frac{W_1 z_1 + W_2 z_2}{W_1 + W_2} = \frac{W_1(\mathrm{p}_1 \mathrm{x} + \mathrm{q}_1 \mathrm{y} + \mathrm{r}_1) + W_2(\mathrm{p}_2 \mathrm{x} + \mathrm{q}_2 \mathrm{y} + r_2)}{W_1 + W_2} \tag{24}$$

Therefore, given any input pair (x, y), the fuzzy system will generate an output using the designed rule base. It should be noted that any t-norm operator can be used in place of the product operator. It is the designer's decision on which t-norm to use. Two commonly used t-norm operators are the minimum and product t-norm.

3.2 Artificial Neural Networks

Neural networks are system representations intended to model the human brain. These networks consist of neurons (or nodes) connected with links. Each link has a corresponding weight and each neuron has an activation function. Activation functions are similar to fuzzy membership functions however the activation functions' parameters remain unchanged. When training the neural network only the link weights are optimized while activation functions are constant. Most commonly, the sigmoid activation function is used for the neurons. To train the network, input data is presented to the system and each node output is calculated until the final output layer is reached. After the total network output is calculated, the error is calculated based on the desired system output and the error signal is then propagated backward toward the input in order to update the link weights using gradient descent. This method for link weight update is called back propagation. [2] provides a detailed derivation of the back propagation learning algorithm.

The sigmoid function is defined as:

$$f(x) = \frac{1}{1+e^{-x}} \tag{25}$$

Taking the derivative with respect to x gives:

$$f'(x) = \frac{e^{-x}}{(1+e^{-x})^2} \tag{26}$$

Representing the derivative in terms of the function yields:

$$f'(x) = f(1 - f) \tag{27}$$

This representation proves useful in calculating the error signals at each layer [2]. A general neural network representation is shown in the figure below:

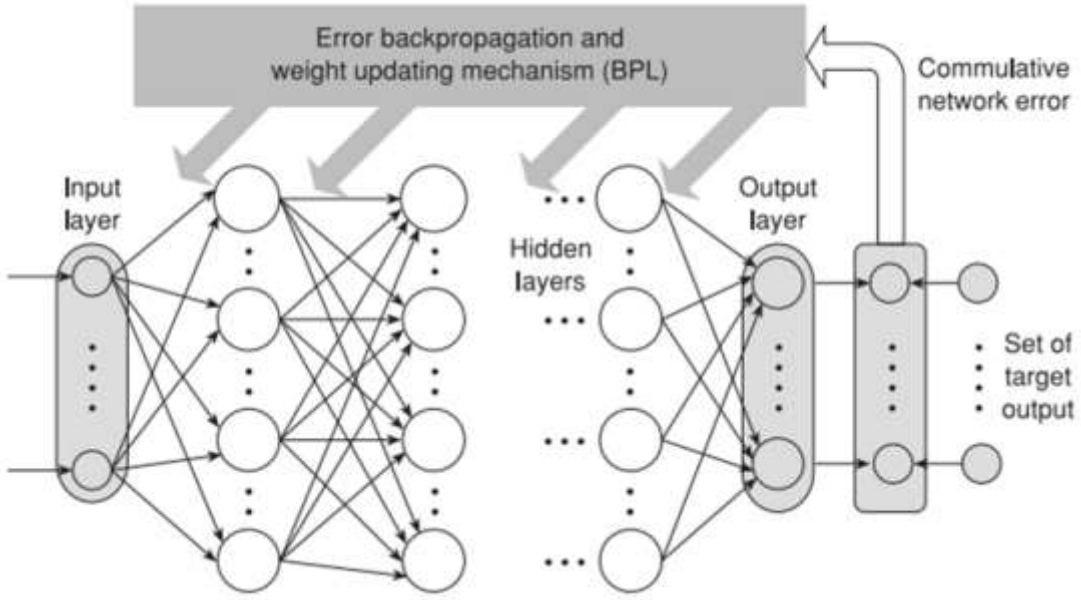

Fig. 3-2: General Neural Network [2].

Fig. 3-2 shows a multi-layer perceptron neural network model. Each circle is a neuron with an activation function and each arrow connecting a neuron in one layer to a neuron in a succeeding layer is the link. There is one input layer, one output layer and N hidden layers. As previously discussed, the link weights are updated by the back propagation algorithm.

If only one perceptron is used, there are only link weights between the input and the neuron. The single perceptron generally uses the signum function to calculate the perceptron's output and is then compared with the target. If each input data pair in a given epoch produces an output that is equal to the target, the training stops otherwise link weights are updated. Training continues until all input data pairs have been classified.

3.3 Adaptive Neuro-Fuzzy Inference System

ANFIS is a system that uses benefitting features of both fuzzy logic and neural networks. It was developed by Jang in the early 1990's [1]. An ANFIS is a fuzzy system that uses neural network training methods to optimize the membership function parameters and the consequent function parameters. As such, ANFIS proves to be a useful system in artificial intelligence based applications. As shown in the figure below, fuzzy logic and neural networks have advantages and disadvantages. By combining both systems into a new system (ANFIS), the ANFIS takes benefits from the advantages of both. Table 3-1 outlines the features of fuzzy logic systems and neural networks of which ANFIS benefits from.

Table 3-1: Fuzzy logic and neural network features [2]

	Fuzzy logic	Neural networks
Representation	Linguistic description of knowledge	Knowledge distributed within computational units
Adaptation	Some adaptation	Adaptive
Knowledge representation	Explicit and easy to interpret	Implicit and difficult to interpret
Learning	Non-existent	Excellent tools for imparting learning
Verification	Easy and efficient	Not straightforward

The general structure of a neuro-fuzzy system is shown in Fig. 3-3 [2].

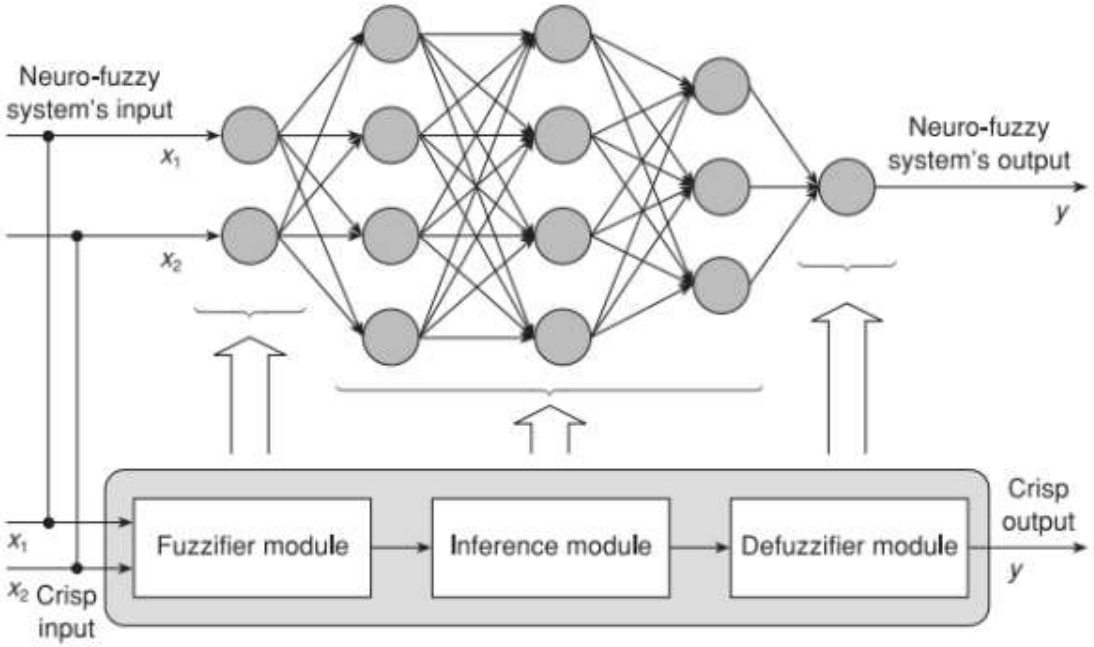

Fig. 3-3: General neuro-fuzzy system.

The neuro-fuzzy system in the above figure is a fuzzy system that has been transformed into a parallel network arrangement. Each layer in the network has its own function as related to the fuzzy system. By arranging the fuzzy system in network representation, neural network training can be applied.

A Sugeno-based adaptive neuro-fuzzy inference system is a fuzzy system that uses neural network based training algorithms to optimize the nonlinear membership parameters and linear consequent parameters. As such, ANFIS is considered to be an intelligent system as there is a type of system learning involved. Instead of the general fuzzy system representation, the fuzzy system can be transformed into a five layer network representation as shown in Fig. 3-4. For this representation, there are two inputs with two membership functions per input and one output. There are also four rules used in this ANFIS. Here, the output of layer 1 is the membership function's value at the specified input (fuzzification). The output of layer 2 is the firing strengths corresponding to each rule. The output of layer 3 is the normalized firing strengths. The output of layer 4 is the final output of each rule. The output of layer 5 is the defuzzified overall system output.

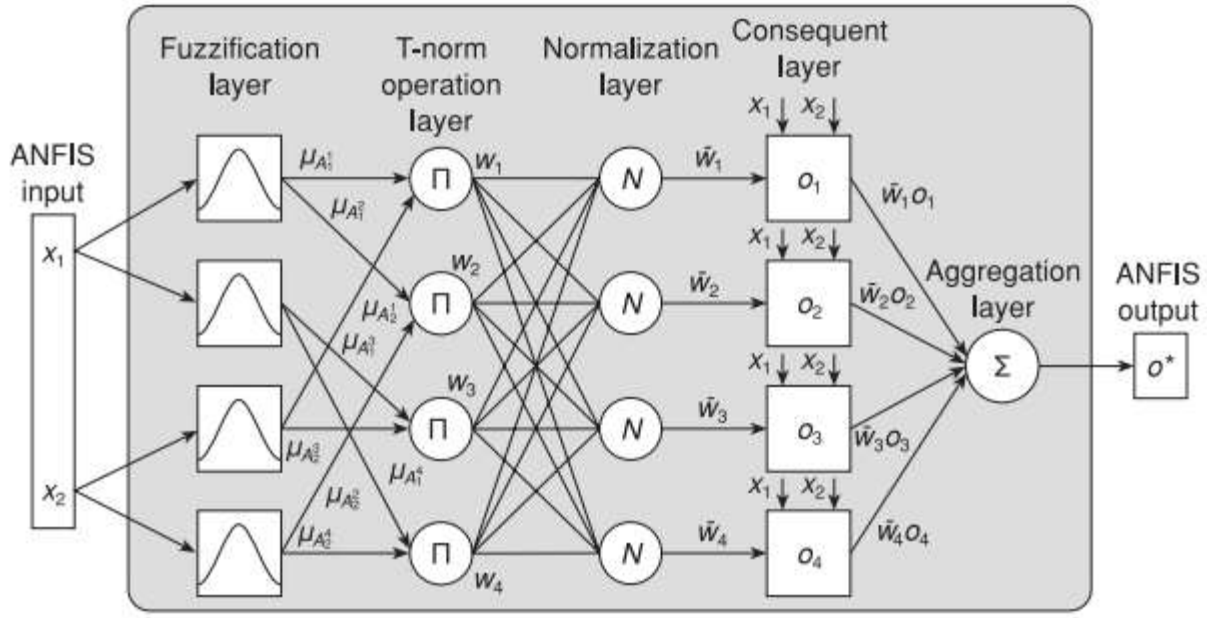

Fig. 3-4: Five layer ANFIS [2].

Using knowledge of Sugeno fuzzy reasoning, the outputs of each layer can be written as follows. Note that neural network training methods are only used to optimize the parameters in layer 1 and in layer 4.

The outputs of the fuzzification layer (layer 1) are:

$$O_{11} = \mu_{A_1}^1 = \mu_{A_1}^2 = \mu_{A1}(x_1) \tag{28}$$

$$O_{12} = \mu_{A_1}^3 = \mu_{A_1}^4 = \mu_{A2}(x_1) \tag{29}$$

$$O_{21} = \mu_{A_2}^1 = \mu_{A_2}^2 = \mu_{B1}(x_2) \tag{30}$$

$$O_{22} = \mu_{A_2}^3 = \mu_{A_2}^4 = \mu_{B2}(x_2) \tag{31}$$

where, μ_{A1} and μ_{A2} are the membership functions for the first input

μ_{B1} and μ_{B2} are the membership functions for the second input

The input membership functions are chosen to be Gaussian functions as follows:

$$\mu_{A1}(x_1) = e^{-0.5\left(\frac{x_1-c_1}{\sigma_1}\right)^2} \tag{32}$$

$$\mu_{A2}(x_1) = e^{-0.5\left(\frac{x_1-c_2}{\sigma_2}\right)^2} \tag{33}$$

$$\mu_{B1}(x_2) = e^{-0.5\left(\frac{x_2-c_3}{\sigma_3}\right)^2} \tag{34}$$

$$\mu_{B2}(x_2) = e^{-0.5\left(\frac{x_2-c_4}{\sigma_4}\right)^2} \tag{35}$$

where, (c_i, σ_i), i=1, 2, 3, 4 are the nonlinear parameters

The outputs of the t-norm operation layer (using product t-norm) are:

$$W_1 = O_{11}O_{21} \tag{36}$$

$$W_2 = O_{11}O_{22} \tag{37}$$

$$W_3 = O_{12}O_{21} \tag{38}$$

$$W_4 = O_{12}O_{22} \tag{39}$$

The outputs of the normalization layer are:

$$\overline{W_1} = \frac{W_1}{W_1+W_2+W_3+W_4} \tag{40}$$

$$\overline{W_2} = \frac{W_2}{W_1+W_2+W_3+W_4} \tag{41}$$

$$\overline{W_3} = \frac{W_3}{W_1+W_2+W_3+W_4} \tag{42}$$

$$\overline{W_4} = \frac{W_4}{W_1+W_2+W_3+W_4} \tag{43}$$

The outputs of the consequent layer are shown in Fig. 3-5.

Here,

$$o_1 = p_1 x_1 + q_1 x_2 + r_1 \quad (44)$$

$$o_2 = p_2 x_1 + q_2 x_2 + r_2 \quad (45)$$

$$o_3 = p_3 x_1 + q_3 x_2 + r_3 \quad (46)$$

$$o_4 = p_4 x_1 + q_4 x_2 + r_4 \quad (47)$$

The output of the aggregation layer is:

$$o^* = \overline{W_1} o_1 + \overline{W_2} o_2 + \overline{W_3} o_3 + \overline{W_4} o_4 \quad (48)$$

With this representation, the training algorithms discussed in the next section can be applied. Nodes in the first and fourth layers contain the nonlinear and linear parameters respectively and hence are referred to as adaptive nodes. All other nodes in the other layers have no parameters and hence are fixed nodes [1].

3.3.1 ANFIS Training

This section will provide the two types of training algorithms used to train the general Sugeno-based ANFIS and will propose a new hybrid training method that implements particle swarm optimization. The general ANFIS uses a hybrid algorithm in which the forward pass is used to optimize the linear parameters and the backward pass is used to update the nonlinear parameters. It is important to note that ANFIS has no link weights or neurons. Thus the back propagation learning algorithm for link weight update cannot be applied to ANFIS training. See the table below for the process of hybrid learning in ANFIS.

Table 3-2: Hybrid learning algorithm passes [1]

	Forward pass	Backward pass
Premise parameters	Fixed	Gradient descent
Consequent parameters	Least-squares estimator	Fixed
Signals	Node outputs	Error signals

3.3.1.1 Recursive Least Squares Estimator

The recursive least-squares estimator (RLSE) [1] is used to optimize the linear parameters in the consequent part of the Sugeno fuzzy rules. The RLSE tries to solve the problem y=Aθ by using training data pairs to estimate the parameters located in θ. Since on-line training is performed, the recursive estimator is required. If offline (or batch) learning is required, then the basic least-squares estimator (LSE) can be used.

The least-squares estimator for batch training will be derived. Then the recursive least-squares estimator will be derived from the LSE equations. The least-squares estimator requires all training data to be inputted at the same time whereas the recursive LSE only requires one training data pair at a time so optimization can be performed as more training data becomes available. As such, LSE is an offline batch learning algorithm whereas RLSE is an online training algorithm and can be performed on a running system as new data becomes available. RLSE is also used in the hybrid training algorithm for ANFIS training.

Consider a system with input $\vec{u}$ with $\vec{\theta}$ containing the linear parameters to be optimized

The system output is defined as:

$$y = \theta_1 f_1(\vec{u}) + \theta_2 f_2(\vec{u}) + \cdots + \theta_n f_n(\vec{u}) \tag{49}$$

Now assume training data is gathered from the system.

Training data pairs are formed as $(\vec{u}_1, y_1), (\vec{u}_2, y_2), \ldots, (\vec{u}_i, y_i), \ldots, (\vec{u}_m, y_m)$

where, y is the desired output of the system.

Consider the output of the system for each training data pair,

$$y_1 = \theta_1 f_1(\overrightarrow{u_1}) + \theta_2 f_2(\overrightarrow{u_1}) + \cdots + \theta_n f_n(\overrightarrow{u_1})$$

$$y_2 = \theta_1 f_1(\overrightarrow{u_2}) + \theta_2 f_2(\overrightarrow{u_2}) + \cdots + \theta_n f_n(\overrightarrow{u_2})$$

.
.
.

$$y_m = \theta_1 f_1(\overrightarrow{u_m}) + \theta_2 f_2(\overrightarrow{u_m}) + \cdots + \theta_n f_n(\overrightarrow{u_m}) \tag{50}$$

In matrix notation, the output becomes,

$$\begin{bmatrix} f_1(\overrightarrow{u_1}) & f_2(\overrightarrow{u_1}) & \dots & f_i(\overrightarrow{u_1}) & \dots & f_n(\overrightarrow{u_1}) \\ f_1(\overrightarrow{u_2}) & f_2(\overrightarrow{u_2}) & \dots & f_i(\overrightarrow{u_2}) & \dots & f_n(\overrightarrow{u_2}) \\ \vdots & \vdots & & \vdots & & \vdots \\ f_1(\overrightarrow{u_\iota}) & f_2(\overrightarrow{u_\iota}) & \cdots & f_i(\overrightarrow{u_\iota}) & \dots & f_n(\overrightarrow{u_\iota}) \\ \vdots & \vdots & & \vdots & & \vdots \\ f_1(\overrightarrow{u_m}) & f_2(\overrightarrow{u_m}) & \cdots & f_i(\overrightarrow{u_m}) & \dots & f_n(\overrightarrow{u_m}) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_i \\ \vdots \\ \theta_n \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_m \end{bmatrix} \tag{51}$$

$$A\vec{\theta} = \vec{y} \tag{52}$$

where,

$$A = \begin{bmatrix} f_1(\overrightarrow{u_1}) & f_2(\overrightarrow{u_1}) & \dots & f_i(\overrightarrow{u_1}) & \dots & f_n(\overrightarrow{u_1}) \\ f_1(\overrightarrow{u_2}) & f_2(\overrightarrow{u_2}) & \dots & f_i(\overrightarrow{u_2}) & \dots & f_n(\overrightarrow{u_2}) \\ \vdots & \vdots & & \vdots & & \vdots \\ f_1(\overrightarrow{u_\iota}) & f_2(\overrightarrow{u_\iota}) & \cdots & f_i(\overrightarrow{u_\iota}) & \dots & f_n(\overrightarrow{u_\iota}) \\ \vdots & \vdots & & \vdots & & \vdots \\ f_1(\overrightarrow{u_m}) & f_2(\overrightarrow{u_m}) & \cdots & f_i(\overrightarrow{u_m}) & \dots & f_n(\overrightarrow{u_m}) \end{bmatrix}, \quad \theta = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_i \\ \vdots \\ \theta_n \end{bmatrix}, \quad y = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_i \\ \vdots \\ y_m \end{bmatrix}$$

Rewriting A gives,

$$A = \begin{bmatrix} \overrightarrow{a_1}^T \\ \overrightarrow{a_2}^T \\ \vdots \\ \overrightarrow{a_\iota}^T \\ \vdots \\ \overrightarrow{a_m}^T \end{bmatrix} \tag{53}$$

Rearranging $\vec{y} = A\vec{\theta}$ gives, $\vec{\theta} = A^+\vec{y}$ (54)

Now, the error between the actual system output and the desired output is defined as,

$$\vec{e} = \vec{y} - A\vec{\theta} \tag{55}$$

Usually there are more training data pairs than there are parameters to be optimized. Thus,

$$m \gg n$$

The objective function is defined as,

$$E(\vec{e}) = (y_1 - \vec{a}_1^T\vec{\theta})^2 + (y_2 - \vec{a}_2^T\vec{\theta})^2 + \cdots + (y_m - \vec{a}_m^T\vec{\theta})^2$$

$$E(\vec{e}) = \sum_{i=1}^{m} (y_i - \vec{a}_i^T\vec{\theta})^2$$

$$E(\vec{\theta}) = \vec{e}^T\vec{e} = (\vec{y} - A\vec{\theta})^T(\vec{y} - A\vec{\theta}) \tag{56}$$

Using some matrix manipulations, the error function becomes,

$$E(\vec{\theta}) = \left[\vec{y}^T - (A\vec{\theta})^T\right]\left(\vec{y} - A\vec{\theta}\right)$$

$$= \left[\vec{y}^T - \vec{\theta}^T A^T\right]\left(\vec{y} - A\vec{\theta}\right)$$

$$= \vec{y}^T\vec{y} - \vec{y}^T A\vec{\theta} - \vec{\theta}^T A^T\vec{y} + \vec{\theta}^T A^T A\vec{\theta}$$

$$E(\vec{\theta}) = \vec{y}^T\vec{y} - 2\vec{y}^T A\vec{\theta} + \vec{\theta}^T A^T A\vec{\theta} \tag{57}$$

To find θ corresponding to the minimum of the error function, the derivative is found and set to zero,

$$\frac{\partial E(\vec{\theta})}{\partial \vec{\theta}} = \vec{0} = -2(\vec{y}^T A)^T + A^T A\vec{\theta} + \left(\vec{\theta}^T A^T A\right)^T$$

$$= -2A^T\vec{y} + A^T A\vec{\theta} + A^T A\vec{\theta}$$

$$0 = -2A^T\vec{y} + 2A^T A\vec{\theta}$$

$$A^T\vec{y} = A^T A\vec{\theta}$$

$$(A^T A)^{-1}A^T\vec{y} = (A^T A)^{-1}A^T A\vec{\theta}$$

$$\vec{\theta} = (A^T A)^{-1}A^T\vec{y} \tag{58}$$

Using (58), any linear parameters in the system can be estimated by using all of the training data pairs.

To derive the recursive LSE, it is assumed that at each time step there is a new training data set available.

$$\vec{\theta}_k = (A^T A)^{-1} A^T \vec{y} = P_k A^T \vec{y} \tag{59}$$

where,

$$P_k = (A^T A)^{-1}$$

By definition,

$$\vec{\theta}_{k+1} = ({A^T}_{k+1} A_{k+1})^{-1} {A^T}_{k+1} \vec{y}_{k+1} \tag{60}$$

Rewriting $\vec{\theta}_{k+1}$ gives,

$$\vec{\theta}_{k+1} = \left\{ \begin{bmatrix} A \\ \vec{a}^T \end{bmatrix}^T \begin{bmatrix} A \\ \vec{a}^T \end{bmatrix} \right\}^{-1} \begin{bmatrix} A \\ \vec{a}^T \end{bmatrix}^T \begin{pmatrix} \vec{y} \\ y \end{pmatrix} \tag{61}$$

where, y and $\vec{a}^T$ correspond to the new training data pair at time k+1

Define,

$${P_k}^{-1} = A^T A$$

$${P_{k+1}}^{-1} = \left\{ \begin{bmatrix} A \\ \vec{a}^T \end{bmatrix}^T \begin{bmatrix} A \\ \vec{a}^T \end{bmatrix} \right\}$$

$${P_{k+1}}^{-1} = \{A^T A + \vec{a}\vec{a}^T\} = {P_k}^{-1} + \vec{a}\vec{a}^T \tag{62}$$

Now,

$$\vec{\theta}_{k+1} = P_{k+1} [A^T \quad \vec{a}] \begin{bmatrix} \vec{y} \\ y \end{bmatrix}$$

$$= P_{k+1} [A^T \vec{y} + \vec{a} y] \tag{63}$$

Since,

$$\vec{\theta}_k = P_k A^T \vec{y} \tag{64}$$

It is found that,

$$P_k{}^{-1}\vec{\theta}_k = A^T\vec{y} \tag{65}$$

$\vec{\theta}_{k+1}$ becomes,

$$\vec{\theta}_{k+1} = P_{k+1}\left[P_k{}^{-1}\vec{\theta}_k + \vec{a}y\right]$$

$$= P_{k+1}\left[(P_{k+1}{}^{-1} - \vec{a}\vec{a}^T)\vec{\theta}_k + \vec{a}y\right]$$

$$= P_{k+1}\left[P_{k+1}{}^{-1}\vec{\theta}_k - \vec{a}\vec{a}^T\vec{\theta}_k + \vec{a}y\right]$$

$$= \vec{\theta}_k + P_{k+1}\left(\vec{a}y - \vec{a}\vec{a}^T\vec{\theta}_k\right)$$

$$\vec{\theta}_{k+1} = \vec{\theta}_k + P_{k+1}\vec{a}\left(y - \vec{a}^T\vec{\theta}_k\right) \tag{66}$$

And,

$$P_{k+1} = \left(P_k{}^{-1} + \vec{a}\vec{a}^T\right)^{-1} \tag{67}$$

Using matrix inversion lemma $[A + BC]^{-1} = A^{-1} - A^{-1}B(I + CA^{-1}B)^{-1}CA^{-1}$, P_{k+1} becomes,

$$P_{k+1} = P_k - \frac{P_k\vec{a}_{k+1}\vec{a}^T{}_{k+1}P_k}{1+\vec{a}^T{}_{k+1}P_k\vec{a}_{k+1}} \tag{68}$$

(68) is only valid if the denominator is non-singular.

Therefore, the recursive least squares estimator is defined by the two equations below:

$$P_{k+1} = P_k - \frac{P_k\vec{a}_{k+1}\vec{a}_{k+1}^T P_k}{1+\vec{a}_{k+1}^T P_k\vec{a}_{k+1}} \tag{69}$$

$$\theta_{\ k+1} = \theta_k + P_{k+1}\vec{a}_{k+1}(y_{k+1} - \vec{a}_{k+1}^T\theta_k) \tag{70}$$

where,

P_{k+1} is the adaptation matrix; $P_{k+1}\vec{a}_{k+1}$ is the adaptation gain vector

Θ_{k+1} is the estimator and contains an estimate of the optimal linear parameter values

a_{k+1} is a vector containing the coefficients of the linear parameters to be optimized

y_{k+1} is the desired system output; $(y_{k+1} - \vec{a}_{k+1}^{T}\theta_k)$ is the prediction error vector

$P_o = \alpha I$, where α is a large number

Θ_o is usually initialized to a vector of zeros

k=1, 2, …, N

N is the number of training data pairs

Applying the RLSE to the five layer ANFIS model described, the input signals propagate through the network until layer 4 at which point RLSE is applied to obtain an estimate for all the linear parameters in this layer. It is important that while using the least squares estimator, the nonlinear input membership parameters must be held constant.

3.3.1.2 Gradient Descent Algorithm

The method of gradient descent is usually used to optimize the nonlinear input membership function parameters. This method works by finding the derivative of the objective (or error) function to determine the gradient. A simple equation is then applied to update the parameters. A detailed discussion on gradient descent can be found in Jang's book [1]. The error function is defined as:

$$E = \left(Z_{out} - {Z_{out}}^{d}\right)^2 \tag{71}$$

where Z_{out} is the ANFIS output and ${Z_{out}}^{d}$ is the desired system output

Assume in this case that 'a' is a nonlinear parameter to be optimized. Since Z_{out} depends on the nonlinear parameters, the chain rule is used to find the derivative of the error function as follows:

$$\frac{\partial E}{\partial a} = \frac{\partial E}{\partial Zout} * \frac{\partial Zout}{\partial a} \tag{72}$$

With the error function defined as in (71),

$$\frac{\partial E}{\partial a} = 2 * \left(Z_{out} - {Z_{out}}^{d}\right) * \frac{\partial Zout}{\partial a} \tag{73}$$

Once this derivative is found, parameter 'a' can be updated as follows:

$$a_{k+1} = a_k - \eta \frac{\partial E}{\partial a} \tag{74}$$

where, η is the learning rate (or step size)

k is the current step or position

k+1 is the next step or position

To force the predicted output to be as close as possible to the desired output, minimum error is desired. Since the gradient points in the direction of steepest increase, a negative step is used in the parameter update equation to ensure a minimum is found.

For gradient descent, the linear parameters are held constant and the error signal is applied from the last layer and propagated backward until layer 1 at which point the nonlinear parameters can be estimated. The main issue with gradient descent is that the algorithm may 'get stuck' in a local minimum rather than the global minimum and hence will not provide the optimum parameters for the system. More complex methods such as genetic algorithms have been developed that do not require the use of the derivative.

If gradient descent is used for the four rule ANFIS as described in section 3.3 with Gaussian input membership functions, the equations used for training can be derived.

To update parameter c_1 in the first input membership function, the derivative of the error is required:

$$\frac{\partial E}{\partial c_1} = 2 * (x_{out} - {x_{out}}^{d}) * \frac{\partial xout}{\partial c_1} \tag{75}$$

And,

$$x_{out} = \frac{1}{W_1+W_2+W_3+W_4} * (W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4) \tag{76}$$

To find the derivative of x_{out} with respect to the nonlinear parameters, the product or quotient rule can be used. In this case, the product rule is used:

$$\frac{\partial x_{out}}{\partial c_1} = \frac{-1}{(W_1 + W_2 + W_3 + W_4)^2}\left[\frac{\partial W_1}{\partial c_1} + \frac{\partial W_2}{\partial c_1}\right](W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4)$$
$$+ \frac{1}{W_1 + W_2 + W_3 + W_4}\left[\frac{\partial W_1}{\partial c_1} f_1 + \frac{\partial W_2}{\partial c_1} f_2\right]$$

$$= \frac{-1}{(W_1 + W_2 + W_3 + W_4)^2}\left[\frac{x_1 - c_1}{\sigma_1^2}\mu_{A1}\mu_{B1} + \frac{x_1 - c_1}{\sigma_1^2}\mu_{A1}\mu_{B2}\right](W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4)$$
$$+ \frac{1}{W_1 + W_2 + W_3 + W_4}\left[\frac{x_1 - c_1}{\sigma_1^2}\mu_{A1}\mu_{B1} f_1 + \frac{x_1 - c_1}{\sigma_1^2}\mu_{A1}\mu_{B2} f_2\right]$$

$$= \frac{(x_1 - c_1)\mu_{A1}}{\sigma_1^2(W_1 + W_2 + W_3 + W_4)}(\mu_{B1} f_1 + \mu_{B2} f_2 - \mu_{B1} x_{out} - \mu_{B2} x_{out})$$

$$= \frac{(x_1 - c_1)}{\sigma_1^2(W_1 + W_2 + W_3 + W_4)}(W_1 f_1 + W_2 f_2 - W_1 x_{out} - W_2 x_{out})$$

$$= \frac{x_1 - c_1}{\sigma_1^2}(\overline{W_1}(f_1 - x_{out}) + \overline{W_2}(f_2 - x_{out}))$$

$$\frac{\partial x_{out}}{\partial c_1} = \frac{x_1 - c_1}{\sigma_1^2}(O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \tag{77}$$

Therefore,

$$\frac{\partial E}{\partial c_1} = 2 * (x_{out} - {x_{out}}^d) * \frac{x_1 - c_1}{\sigma_1^2} * (O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \tag{78}$$

Relating the derivative to the error function, the following representation of the derivative is obtained:

$$\frac{\partial E}{\partial c_1} = 2 * \sqrt{E} * \frac{x_1 - c_1}{\sigma_1^2} * (O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \tag{79}$$

Similarly the derivatives of the error with respect to the remaining three 'c' parameters can be found as:

$$\frac{\partial E}{\partial c_2} = 2 * \sqrt{E} * \frac{x_1 - c_2}{\sigma_2^2} * (O_3 + O_4 - x_{out}(\overline{W_3} + \overline{W_4})) \tag{80}$$

$$\frac{\partial E}{\partial c_3} = 2 * \sqrt{E} * \frac{x_2 - c_3}{\sigma_3^2} * (O_1 + O_3 - x_{out}(\overline{W_1} + \overline{W_3})) \tag{81}$$

$$\frac{\partial E}{\partial c_4} = 2 * \sqrt{E} * \frac{x_2 - c_4}{\sigma_4^2} * (O_2 + O_4 - x_{out}(\overline{W_2} + \overline{W_4})) \tag{82}$$

With these derivatives, parameter 'c' in each input membership can be updated using:

$$c_{i,k+1} = c_{i,k} - \eta \frac{\partial E}{\partial c_i}, \quad i = 1, 2, 3, 4 \tag{83}$$

Now, to update parameter σ_1 in the first input membership function, the derivative of the error is required. Note that a similar process as parameter 'c' is used to determine the derivative of the error with respect to σ.

$$\frac{\partial E}{\partial \sigma_1} = 2 * (x_{out} - {x_{out}}^d) * \frac{\partial xout}{\partial \sigma_1} \tag{84}$$

And,

$$x_{out} = \frac{1}{W_1+W_2+W_3+W_4} * (W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4) \tag{85}$$

To find the derivative of x_{out} with respect to the nonlinear parameters, the product or quotient rule can be used. In this case, the product rule is used:

$$\frac{\partial x_{out}}{\partial \sigma_1} = \frac{-1}{(W_1 + W_2 + W_3 + W_4)^2}\left[\frac{\partial W_1}{\partial \sigma_1} + \frac{\partial W_2}{\partial \sigma_1}\right](W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4) + \frac{1}{W_1 + W_2 + W_3 + W_4}\left[\frac{\partial W_1}{\partial \sigma_1} f_1 + \frac{\partial W_2}{\partial \sigma_1} f_2\right]$$

$$= \frac{-1}{(W_1 + W_2 + W_3 + W_4)^2}\left[\frac{(x_1 - c_1)^2}{\sigma_1^3}\mu_{A1}\mu_{B1} + \frac{(x_1 - c_1)^2}{\sigma_1^3}\mu_{A1}\mu_{B2}\right](W_1 f_1 + W_2 f_2 + W_3 f_3 + W_4 f_4) + \frac{1}{W_1 + W_2 + W_3 + W_4}\left[\frac{(x_1 - c_1)^2}{\sigma_1^3}\mu_{A1}\mu_{B1} f_1 + \frac{(x_1 - c_1)^2}{\sigma_1^3}\mu_{A1}\mu_{B2} f_2\right]$$

$$= \frac{(x_1 - c_1)^2 \mu_{A1}}{\sigma_1^3 (W_1 + W_2 + W_3 + W_4)}(\mu_{B1} f_1 + \mu_{B2} f_2 - \mu_{B1} x_{out} - \mu_{B2} x_{out})$$

$$= \frac{(x_1 - c_1)^2}{\sigma_1^3 (W_1 + W_2 + W_3 + W_4)}(W_1 f_1 + W_2 f_2 - W_1 x_{out} - W_2 x_{out})$$

$$= \frac{(x_1 - c_1)^2}{\sigma_1^3}(\overline{W_1}(f_1 - x_{out}) + \overline{W_2}(f_2 - x_{out}))$$

$$\frac{\partial x_{out}}{\partial \sigma_1} = \frac{(x_1 - c_1)^2}{\sigma_1^3}(O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \tag{86}$$

Therefore,

$$\frac{\partial E}{\partial \sigma_1} = 2 * (x_{out} - {x_{out}}^d) * \frac{(x_1 - c_1)^2}{\sigma_1^3} * (O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \quad (87)$$

Relating the derivative to the error function, the following representation of the derivative is obtained:

$$\frac{\partial E}{\partial \sigma_1} = 2 * \sqrt{E} * \frac{(x_1 - c_1)^2}{\sigma_1^3} * (O_1 + O_2 - x_{out}(\overline{W_1} + \overline{W_2})) \quad (88)$$

Similarly the derivatives of the error with respect to the remaining three 'σ' parameters can be found as:

$$\frac{\partial E}{\partial \sigma_2} = 2 * \sqrt{E} * \frac{(x_1 - c_2)^2}{\sigma_2^3} * (O_3 + O_4 - x_{out}(\overline{W_3} + \overline{W_4})) \quad (89)$$

$$\frac{\partial E}{\partial \sigma_3} = 2 * \sqrt{E} * \frac{(x_2 - c_3)^2}{\sigma_3^3} * (O_1 + O_3 - x_{out}(\overline{W_1} + \overline{W_3})) \quad (90)$$

$$\frac{\partial E}{\partial \sigma_4} = 2 * \sqrt{E} * \frac{(x_2 - c_4)^2}{\sigma_4^3} * (O_2 + O_4 - x_{out}(\overline{W_2} + \overline{W_4})) \quad (91)$$

With these derivatives, parameter 'σ' in each input membership can be updated using:

$$\sigma_{i,k+1} = \sigma_{i,k} - \eta \frac{\partial E}{\partial \sigma_i}, \quad i = 1, 2, 3, 4 \quad (92)$$

3.3.1.3 Proposed Hybrid Training Method

Since the basic training method using LSE and gradient descent (GD) has many problems, a new hybrid training method is proposed. The proposed algorithm uses RLSE for linear

consequent parameter training and particle swarm optimization (PSO) for nonlinear membership function parameter training. To the author's knowledge, RLSE-PSO training has not been applied to a solar photovoltaic energy conversion system using a buck boost converter for MPPT. The proposed algorithm is used to train the designed ANFIS to produce a duty cycle for the converter in order to deliver maximum power from the panel to the load at all times regardless of irradiance conditions.

The proposed hybrid training algorithm proves more beneficial to ANFIS training, when compared to LSE-GD, as no derivative of the objective function is required. Thus, if the derivative does not exists or there are discontinuities in the objective function, the proposed algorithm can still find the optimal parameters corresponding to the minimum of the objective function. Further, PSO has the ability to search for the global minimum of the objective function and avoid any local minima. By changing two parameters (the personal and global learning rates) in the PSO algorithm, the global optimum can be found.

The designed ANFIS with RLSE-PSO training is used as the maximum power point tracking controller for the solar panel. The proposed training algorithm uses system data obtained while running the perturb and observe MPPT algorithm. Training the ANFIS will allow the MPPT controller to effectively calculate the duty cycle for the converter corresponding to the maximum power point based on any given panel output voltage and current within the specified region of operation. Maximum power point tracking is discussed in the next chapter. Table 3-2 below shows the process of the proposed hybrid learning algorithm.

Table 3-3: Proposed hybrid learning algorithm passes

	Forward Pass	**Backward Pass**
Premise parameters	Fixed	Particle swarm optimization
Consequent parameters	Least-squares estimator	Fixed
Signals	Node outputs	Error signals

3.3.1.3.1 Particle Swarm Optimization

This section provides a discussion on particle swarm optimization and addresses some of the benefits of implementing it for optimization purposes. Further, its relation to ANFIS training will also be explored.

Particle swarm optimization is an evolutionary algorithm that is modeled after the behaviour of a flock of birds [6] or a school of fish. In the PSO algorithm, a random population is set with a number of members (called particles). Each member in the population contains a solution to the given problem. Imagine a flock of birds. Each bird in the flock moves according to its velocity and position and is also influenced by the other birds' position. If the flock is searching for food, one bird in the flock will have the position best suited to find the food (the global best). All other birds will change their velocity and position based on this globally best position and their own best position in order to reach the food. As a result, the swarm (or flock) moves towards the best solution for a given problem. Therefore, by using PSO, the global minimum of the objective function is usually found depending on the initial positions of the members in the population and improves the tracking of the maximum power point as compared to conventional MPPT methods.

The PSO algorithm is outlined below [4]. The population originally consists of randomly generated members. Each particle in the population has its own current position and best position called p_{best}. The globally best solution (called g_{best}) is the best solution among all of the particle's p_{best}. The goal of the PSO algorithm is to move the particles towards their p_{best} and the overall g_{best}.

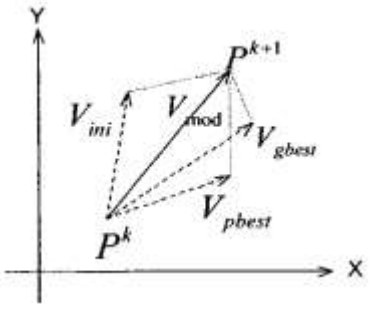

Fig. 3-5: Particle position update [4].

Fig. 3-5 above shows the process of updating a particle's position using PSO [4]. P^k is the particles current position, P^{k+1} is the particle's new update position, V_{ini} is the initial velocity of the particle, V_{mod} is the new velocity, V_{pbest} and V_{gbest} show the velocity due to p_{best} and g_{best} respectively. The particle swarm optimization algorithm is as follows:

1. Initialize a population of M particles with randomly generated positions and velocities
2. For each particle, compute its fitness value according to the optimization problem
3. Compare each particle's current fitness value with its best fitness value (p_{best}'s fitness). If the current value is less than the value from p_{best}, update p_{best} with the current position.
4. Compare each particle's current fitness value with the best fitness value among all particle (g_{best}'s fitness). If the current value is less than the value from g_{best}, update g_{best} with the current position.
5. Update each particle's velocity and position according to the equations below
6. Repeat steps 2-5 until a desired fitness level is achieved or a defined number of iterations is reached.

$$v = w * v + c_1 * rand(0,1) * (p_{best} - x) + c_2 * rand(0,1) * (g_{best} - x) \quad (93)$$

$$x = x + v \quad (94)$$

where,

v is the velocity of each particle

w is an inertia weight that reduces the effect of the previous velocity as the search progresses

c_1 and c_2 are the local and global learning rates respectively. Generally $c_1=c_2$ [38]

$rand(0,1)$ is a random number between 0 and 1

g_{best} is the best position among all members (globally best position)

p_{best} is the best position for a given member (personal best position)

x is the current position of the particle

The choice of the parameters in PSO can significantly affect the algorithm's ability to perform optimization effectively. Therefore, care must be taken in order for PSO to have the best chance at finding the optimal solution for a problem. Trial and error method can be used to determine: the ideal number of particles in the population, the inertia coefficient, the personal and global learning rates and the limits on the particle's positions. [4] provides an explanation on how each of the parameters affects the searching ability of the PSO algorithm. It is important to note that limits should be placed on the generated solutions in order to avoid the particle's movements outside the search space. If the position of a particle exceeds the limits, its value is set to the limits and the search continues. Fig. 3-6 below shows a flow chart outlining the PSO algorithm.

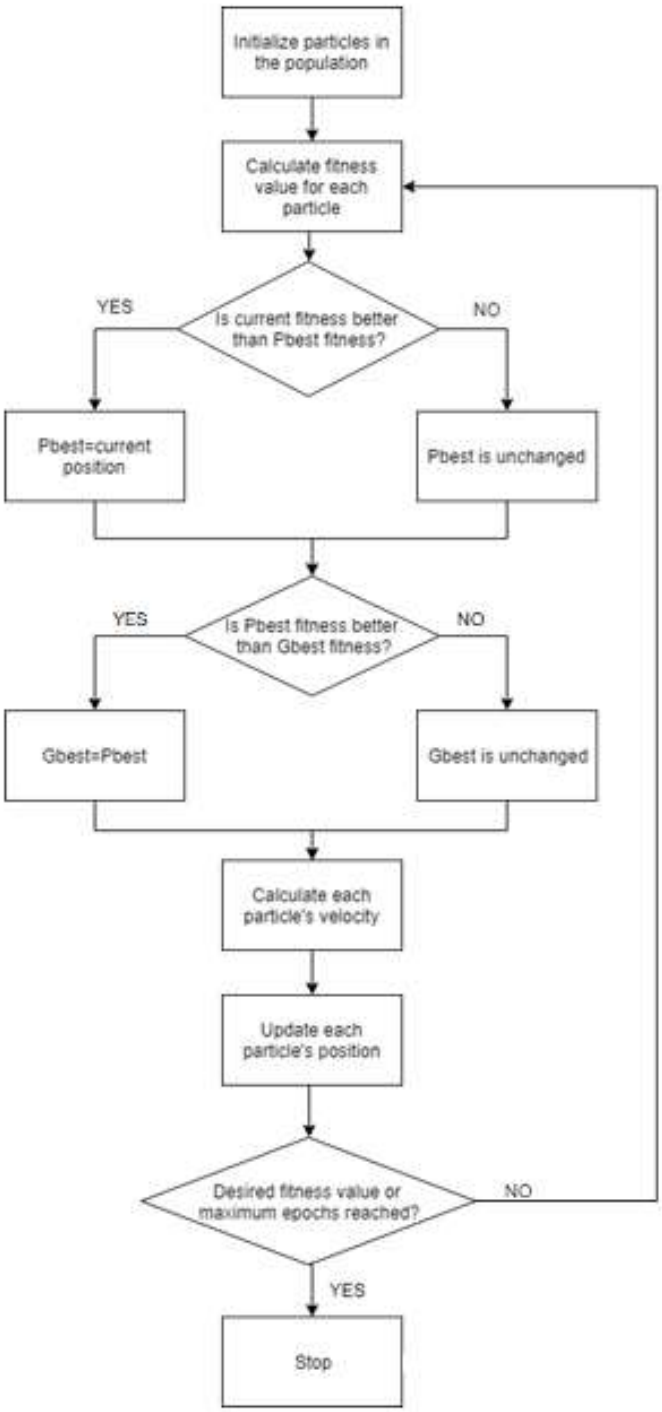

Fig. 3-6: PSO algorithm.

For the proposed RLSE-PSO algorithm applied to the designed ANFIS, the nonlinear input membership function parameters are to be optimized. Thus, each particle in the population contains values of these parameters corresponding to a possible solution that minimizes the error

function. In this case, the error (or cost) function is defined as the root mean squared difference between the desired system output and the actual calculated system output (i.e. the root mean square error). For the purposes of this thesis, the system output is chosen as the duty cycle applied to the MOSFET of the buck-boost converter. Each particle is generated as an array with the center and standard deviation of the function as the columns and each row is associated with one membership function. For example, if there are two membership functions with two parameters per function to be optimized, each particle will be a 2x2 matrix where position (1,1) is the center of the first membership function, position (1,2) is the standard deviation of the first membership function, position (2,1) is the center of the second membership function and position (2,2) is the standard deviation of the second membership function. The proposed algorithm can be used to train any ANFIS system where both linear and nonlinear parameters exist. If the Gaussian membership function is not used, the RLSE-PSO algorithm can still be used; only minor modifications to the Matlab program are required The RLSE trains the linear parameters and the PSO trains the nonlinear parameters.

Chapter 4

Maximum Power Point Tracking and Inverter Control Theory

This chapter delivers a discussion on the maximum power transfer theorem and the perturb and observe (P&O) MPPT algorithm. It also provides inverter control and filter theory as related to the scope of the thesis.

4.1 Maximum Power Transfer Theorem

It is desirable to determine how to transfer maximum power from the source to the load for various applications. The case where the load is purely resistive will be considered and an expression derived that guarantees maximum power transfer.

Consider a simple series circuit as shown in Fig. 4-1 with a voltage source, source resistance and load resistance. The goal of this derivation is to determine the relationship between the source and load resistances that guarantee maximum power will be transferred to the load.

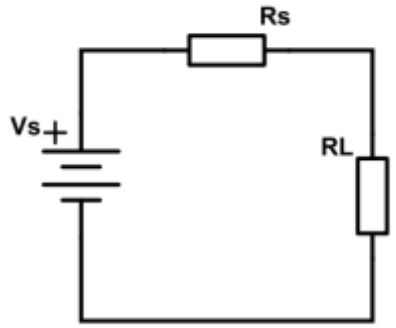

Fig. 4-1: Series circuit with resistive load.

An expression can be found for the current flowing in the circuit and the power dissipated in the load:

$$I = \frac{V_S}{R_L + R_S} \tag{95}$$

$$P_{load} = \left(\frac{V_S}{R_L + R_S}\right)^2 R_L \tag{96}$$

Expanding and simplifying gives,

$$P_{load} = \frac{V_S^2}{\frac{R_S^2}{R_L} + 2R_S + R_L} \tag{97}$$

The maximum load power occurs when P_{load} is maximized. P_{load} is maximized when the derivative of the denominator with respect to R_L is equal to zero.

Taking the derivative of the denominator of (97) with respect to R_L gives,

$$\frac{d(\frac{R_S^2}{R_L} + 2R_S + R_L)}{dR_L} = -\frac{R_S^2}{R_L} + 1 \tag{98}$$

Setting (98) equal to zero and solving for R_L,

$$\frac{R_S^2}{R_L} = 1$$

$$R_L = \pm R_S \tag{99}$$

Resistance cannot be negative so the positive solution is chosen. Taking the second derivative yields a negative value. Therefore, maximum power is transferred to the load when $R_L = R_S$.

As such, for maximum power transfer from a solar panel to a load, the load resistance must be equal to the source resistance. Due to the nonlinear characteristics of the solar panel, the output resistance of the solar panel varies significantly with changing weather conditions. For this reason, a DC-DC converter is used in order to match the panel's nonlinear output impedance to the load regardless of weather conditions or changes in the load.

4.2 Maximum Power Point Tracking

Maximum power point tracking is used to guarantee maximum power is delivered from the solar panel to the load of the system. Usually the maximum power point tracking is performed by a DC-DC converter but it can also be implemented directly with the inverter. If only the inverter is used for MPPT and three phase generation, the control of the inverter becomes much more complicated but the need for a DC-DC converter is removed. As a result, the system that only uses the inverter for MPPT is less expensive than the system proposed in this thesis; however, the computational cost is greater due to the requirement of a significantly more complex control algorithm. The MPPT algorithm calculates a duty cycle corresponding to the MPP and hence changes the duty cycle that is applied to the DC-DC converter. By changing the duty cycle, the impedance apparent to the solar panel is changed to match the load impedance to the solar panel impedance. For a given temperature and irradiance, there is a solar panel output voltage (V_{mpp}) and current (I_{mpp}) corresponding to the maximum power point. These values give an optimal resistance of $R_{opt} = \frac{V_{mpp}}{I_{mpp}}$. The optimal resistance is the resistance that the solar panel needs to be connected to in order to achieve maximum power output from the panel. Since the load cannot be changed at all times for varying weather conditions, a DC-DC converter is used to match any load resistance to the optimal resistance of the panel. By this application, the load will always be matched to the changing optimal solar panel resistance. As such, the DC-DC converter is used as an impedance matcher for MPPT purposes. Thus, the DC-DC converter can effectively match the impedance of the load to the solar panel. A buck boost converter or single ended primary inductor converter (SEPIC) are two desirable converters for maximum power point tracking as they both can step up or step down an input voltage. As a result, both converters can track the maximum power point regardless of weather or load conditions. If the load changes, the converter is still able to match the solar panel output impedance to the load. Assume the temperature, irradiance or the load changes, the MPPT algorithm will recalculate a duty cycle that will allow the converter to achieve load matching and draw maximum power from the solar panel. For the purposes of this thesis, the temperature is assumed to remain constant at all times (constant solar panel voltage at the MPP) and the irradiance is allowed to change (changing solar panel output current). If changes in temperature are to be included, the training data set becomes

larger and the training will take more resources and computational time. The designed training algorithm for the ANFIS can easily be applied if the temperature is allowed to change by gathering more data over the allowed temperature and irradiance operating range.

Further, it is not necessary to design the system to operate at temperatures below 0°C as these temperatures occur in winter months. As such, the solar irradiance is much lower than on a spring or summer day so the solar panels do not work well (unless the panels are not snow covered; then the albedo effect helps). Also, when the temperatures are cold, snow usually covers the solar panels blocking most solar irradiation from reaching the panel. For these reasons, the panels do not generate enough energy to require operation at negative temperatures.

4.2.1 Perturb and Observe Algorithm

The perturb and observe MPPT algorithm is one of the simplest MPPT algorithms to understand and implement. For this reason, it is the most widely used MPPT algorithm in industry today. This algorithm perturbs the duty cycle (it either increases or decreases the duty cycle) and observes the effect on the power generated by the solar panel. Depending on the change in power and change in voltage, the duty cycle is increased or decreased accordingly. The main disadvantage of the P&O algorithm is the inherent oscillations around the maximum power point. These oscillations result in a loss of available power and hence decrease the efficiency of the solar panel. Thus, it is desirable to develop new MPPT algorithms that provide a more stable response at the MPP. The operation of the P&O algorithm is shown in Fig. 4-2 below.

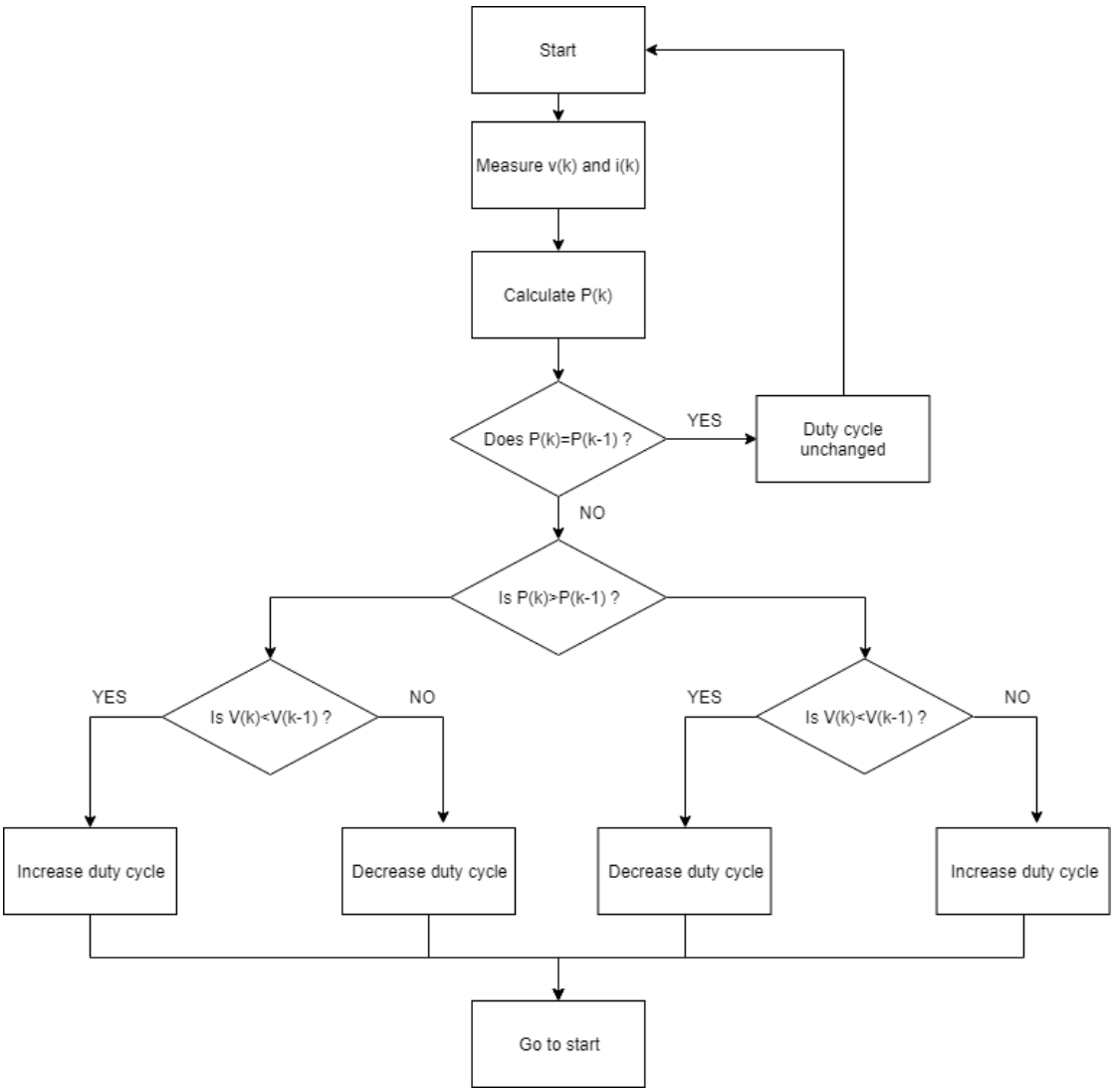

Fig. 4-2: Perturb and Observe MPPT Algorithm.

4.3 Two Level Voltage Source Inverter Control

This section provides a brief analysis of a voltage source inverter, outlines the Park transform and the sinusoidal PWM control method as well as the closed loop dq axis-based voltage control for a two level voltage source inverter.

4.3.1 Park Transform

The Park transform converts a time varying three phase signal into two dc components in the rotating reference frame. The inverse Park transform converts the dq0 components in the rotating reference frame into the corresponding abc components in time domain. To find the inverse Park transform, the inverse of the 3x3 matrix in (100) below must be found. To convert the abc signals to dq0 components or to perform the inverse operation, Simulink has a built in block that will perform these calculations. For implementation purposes, the abc to dq0 transformation is not available as a Simulink block and hence the conversion must be realized as a subsystem and the associated equations drawn.

When the rotating frame is aligned with the 'a' axis, the Park transform is defined by,

$$\begin{bmatrix} V_d \\ V_q \\ V_0 \end{bmatrix} = \frac{2}{3}\begin{bmatrix} \cos(\theta) & \cos(\theta - {}^{2\pi}/_{3}) & \cos(\theta + {}^{2\pi}/_{3}) \\ -\sin(\theta) & -\sin(\theta - {}^{2\pi}/_{3}) & -\sin(\theta + {}^{2\pi}/_{3}) \\ {}^{1}/_{2} & {}^{1}/_{2} & {}^{1}/_{2} \end{bmatrix}\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{100}$$

The inverse Park transform for the same case is defined by,

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 1 \\ \cos(\theta - {}^{2\pi}/_{3}) & -\sin(\theta - {}^{2\pi}/_{3}) & 1 \\ \cos(\theta + {}^{2\pi}/_{3}) & -\sin(\theta + {}^{2\pi}/_{3}) & 1 \end{bmatrix}\begin{bmatrix} V_d \\ V_q \\ 0 \end{bmatrix} \tag{101}$$

where,

V_a, V_b, V_c are the balanced three phase signals in the time varying abc reference frame

V_d, V_q are the dc signals in the rotating reference frame

θ is the angle between phase 'a' in the time varying reference frame and the q axis in the rotating reference frame

4.3.2 Sinusoidal PWM

The section discusses the sinusoidal PWM method to control a three phase voltage source inverter. The circuit configuration and the output waveforms are provided along with a mathematical expression for the output waveforms. A controller can be used to generate the reference control signals ($V_{contol,A}$, $V_{control,B}$ and $V_{control,C}$) and hence closed loop control is established. Fig. 4-3 shows a three phase inverter [17]. Six switches are used to make three legs of the inverter. A pulse width modulation (PWM) signal is applied to all six switches in order to convert the DC voltage into a three phase AC voltage. A capacitor is used at the input of the inverter to maintain a constant dc link voltage required by the inverter for proper operation.

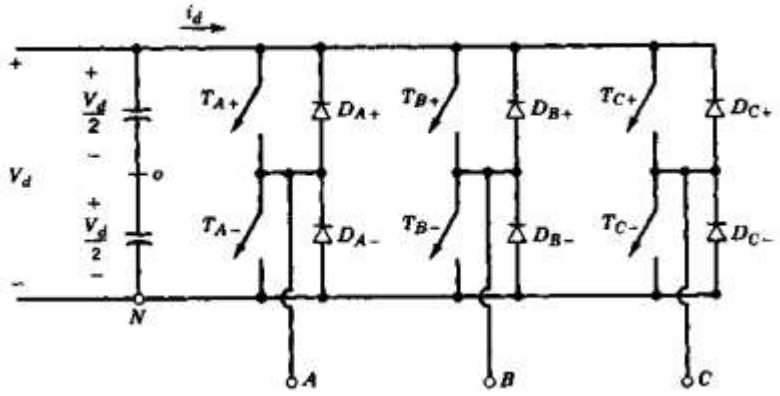

Fig. 4-3: Three phase inverter.

By choosing an appropriate switching scheme, the filtered output of the inverter can be made almost perfectly sinusoidal with a phase shift of 120° between phases. The frequency of the AC waveforms is determined by the switching scheme and the topology of the full system. If the inverter is connected to the grid, the grid frequency can be measured and used in the generation of the PWM signals for control of the inverter. If stand-alone operation is required, more complicated methods are required to control the VSI as the grid frequency is not available and as such, must be generated with the control scheme.

Consider the perfect case in which only the fundamental component of the output waveforms exist. For abc sequence, the phase voltages (line to neutral) can be expressed as:

$$V_{An} = V_m \sin wt \tag{102}$$

$$V_{Bn} = V_m \sin(wt - 120°) \tag{103}$$

$$V_{Cn} = V_m \sin(wt + 120°) \tag{104}$$

where V_m is the peak value of the signal

The line voltages are obtained as:

$$V_{AB} = V_{An} - V_{Bn} \tag{105}$$

$$V_{BC} = V_{Bn} - V_{Cn} \tag{106}$$

$$V_{CA} = V_{Cn} - V_{An} \tag{107}$$

To control the inverter and generate PWM pulses for the gate of the switches using bipolar PWM switching, a reference signal is compared with a triangular signal and hence the PWM generated. Three control signals are required (one for each phase) and are compared to the same triangle waveform. Note that the reference signal is the desired phase voltage output of the inverter for each phase. Fig. 4-4 shows how the comparison of the triangle and reference control signals generate the unfiltered phase voltages at the output of the inverter.

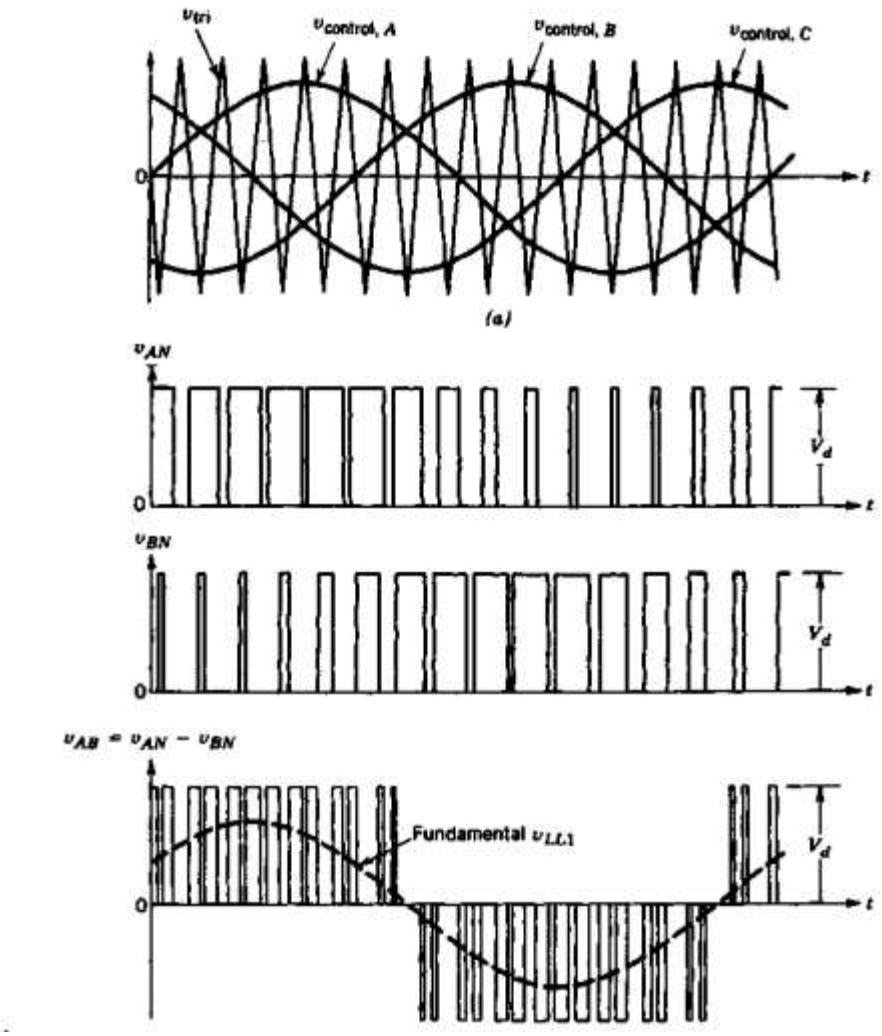

Fig. 4-4: Phase voltage generation [17].

The operation of the PWM generation and application to the inverter switches is as follows:

When $V_{control, k} > V_{tri}$, T_k^+ is on and T_k^- is off
When $V_{control, k} < V_{tri}$, T_k^- is on and T_k^+ is off

where,

k= A, B, C

To prevent over modulation and achieve linear modulation, the amplitude of the control signal must be less than the triangle signal. This fact introduces the amplitude modulation index,

m_a. m_a is defined as the ratio of the peak value of the control signal to the peak value of the triangle signal.

$$m_a = \frac{\hat{V}_{control}}{\hat{V}_{tri}} \tag{108}$$

where, m_a is the amplitude modulation index

$\hat{V}_{control}$ is the peak value of the control signal

$\hat{V}_{tri}$ is the peak value of the triangle signal

Usually m_a is less than or equal to 0.8 [17] which means the peak voltage level of the control signal is at most 80 percent of the peak voltage of the triangle signal. Further, the frequency of the triangle signal must be much higher than the control signal's frequency. The frequency of the triangle signal determines the frequency of the PWM signal. As such, a frequency modulation index is introduced as,

$$m_f = \frac{f_{tri}}{f_{control}} \tag{109}$$

where, m_f is the frequency modulation index

f_{tri} is the frequency of the triangle signal

$f_{control}$ is the frequency of the control signal

It is important that m_f is chosen to be an odd number. If m_f is and odd number then the output signals of the inverter have odd symmetry and half-wave symmetry [17]. Due to these features, the even harmonics are not found in the signals and only the odd harmonics exist. This reduces the THD of the three phase output and allows for simpler filtering. Further, if m_f is chosen as a large integer, the harmonics in the output voltage waveforms are located at high frequencies. As such, filtering the harmonics out of the waveform is easier than if a lower m_f was used. Table 4-1 below shows the generalized harmonics in the phase voltages for a large m_f value.

Table 4-1: Generalized harmonics in the phase voltages for large m_f [17]

h \ m_a	0.2	0.4	0.6	0.8	1.0
1	0.2	0.4	0.6	0.8	1.0
Fundamental					
m_f	1.242	1.15	1.006	0.818	0.601
$m_f \pm 2$	0.016	0.061	0.131	0.220	0.318
$m_f \pm 4$					0.018
$2m_f \pm 1$	0.190	0.326	0.370	0.314	0.181
$2m_f \pm 3$		0.024	0.071	0.139	0.212
$2m_f \pm 5$				0.013	0.033
$3m_f$	0.335	0.123	0.083	0.171	0.113
$3m_f \pm 2$	0.044	0.139	0.203	0.176	0.062
$3m_f \pm 4$		0.012	0.047	0.104	0.157
$3m_f \pm 6$				0.016	0.044
$4m_f \pm 1$	0.163	0.157	0.008	0.105	0.068
$4m_f \pm 3$	0.012	0.070	0.132	0.115	0.009
$4m_f \pm 5$			0.034	0.084	0.119
$4m_f \pm 7$				0.017	0.050

If the switching frequency of the inverter is high (i.e. a large m_f is chosen), then the need for a precise filter is relaxed since the harmonics will be located at high frequencies. As such, an inductor filter would be sufficient to remove harmonics in the phase voltages even though the attenuation of an inductor filter is -20dB/decade over all frequencies. If a smaller m_f value is used, the harmonics are located closer to the fundamental frequency and a better filter must be designed for these harmonics to effectively be removed without affecting the fundamental component. A trade-off between filter complexity and switching noise arises with different m_f values. A low m_f requires a more complex and precise filter to remove the harmonics but the noise generated due to the IGBTs switching is low. For high values of m_f, the filter requirements are relaxed, however, the switching noise is much greater than when a low m_f is used. Therefore, when considering switching noise and harmonics, some care must be taken to satisfy relatively low switching noise and effective harmonic filtering. A more in-depth discussion of inverter control can be found in [17], [18].

4.3.3 d-q Axis Control

This control technique uses the dq axis components of the three phase signals to control the voltage at the output of the inverter. In this thesis, this control method is used for voltage control of the inverter. For grid connection, the voltage of the incoming signals must be slightly higher than the grid voltage. As such, voltage control is required to ensure proper grid connection. In this thesis, no current control is applied to the inverter and hence, the inverter output current can vary according to the dc link. The three phase output voltage of the inverter is measured and a phase locked loop (PLL) is used to determine the angle of the grid for synchronization purposes. Park's transformation is applied to the three phase signals to generate the dq components corresponding to this three phase signal. Next, the d and q axis errors are calculated. The desired d axis voltage is set to the peak value of the desired three phase output voltages and the desired q axis voltage is set to zero. Then, the two error signals are then applied to a PI controller that generates the dq reference voltages for inverter control. The dq reference voltages are then converted into abc reference signals by applying the inverse Park transform. Finally, the abc reference signals are fed into the 2-level PWM generator block and these pulses are then sent to the gates of the transistors in the inverter. By controlling the voltage and achieving the desired peak or root mean square (RMS) value at the output of the inverter, effective grid connection is possible. For simulation purposes, the schematic of the control algorithm can be found in Appendix B.

Here, a three phase reference signal is generated by a PI controller for output voltage control. The q-axis reference voltage is set to zero so the controller forces V_q to be zero. The d-axis reference voltage is set to the desired peak voltage level of the inverter's output voltage. Note that the total reference voltage in the dq frame is defined by the vector sum of the d and q axis voltage components. The magnitude of the reference voltage is defined as,

$$V_{ref} = \sqrt{V_d^2 + V_q^2} \tag{110}$$

By setting the q-axis to zero and the d-axis to the desired peak output voltage level, the reference voltage, V_{ref}, becomes equal to the desired peak output voltage level. This concept proves the validity of the inverter voltage control scheme used in this thesis.

4.4 LC Filter

This section provides the theory and design equations for a LC filter. Fig. 4-5 shows the circuit diagram for the LC filter.

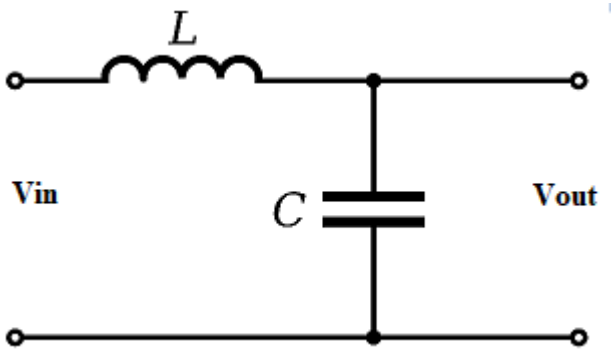

Fig. 4-5: LC Filter.

Note that this filter is for one phase of the three phase inverter output. As such, three LC filters will be connected to filter all three phases of the inverter output. The inductor is responsible for smoothing out the current while the capacitor is responsible for smoothing out the voltage. In the configuration chosen, the filter is a low pass filter with its cut-off frequency determined by the values of the inductor and capacitor. For a system with a fundamental frequency of 60Hz as discussed in this thesis, it is desirable to make the cut-off frequency 60Hz or slightly higher in order to remove higher order harmonics and only keep the fundamental component of the signal. By filtering out the higher order harmonics, the total harmonic distortion of the voltage and current is much less than if the other harmonics were included. The filtering effectively generates an almost purely sinusoidal signal at the output of the filter. Converting the filter into s-domain is shown in Fig. 4-6.

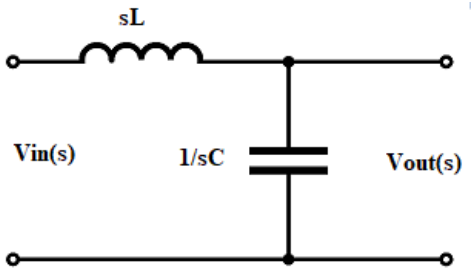

Fig. 4-6: Filter in s-domain.

Referring to Fig. 4-6, Kirchhoff's voltage law (KVL) can be applied and the transfer function of the filter obtained.

$$V_{out}(s) = \frac{V_{in}(s)\left(\frac{1}{sC}\right)}{\frac{1}{sC} + sL}$$

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{LCs^2+1} \tag{111}$$

Note that s=jw, where jwL is the impedance of the inductor and 1/jwC is the impedance of the capacitor. With this transfer function, the frequency response of the filter can be obtained. The frequency response of the filter is discussed in more detail in Chapter 5. Further, the cut-off frequency of the filter is defined as,

$$f_c = \frac{1}{2\pi\sqrt{LC}} \tag{112}$$

This equation can be used to determine the inductor and capacitor values required to achieve a desired cut-off frequency for the filter.

Chapter 5

System Design

This chapter offers a discussion on the design of the proposed solar photovoltaic energy conversion system. The design of the buck boost converter, the RLSE algorithm and the filter is outlined in this section. Note that the PSO algorithm used is the same as the one discussed in Chapter 3. The ANFIS structure used for this thesis is the same as the ANFIS discussed in Chapter 3.

For the simulation, the designed system consists of a solar panel, a buck boost converter controlled with the proposed MPPT algorithm, a two level three phase voltage source inverter with a LC filter, a three phase load and connection to the grid. Fig. 5-1 below shows the general structure of a solar energy conversion system. Fig. 5-2 outlines the solar energy conversion proposed in this thesis.

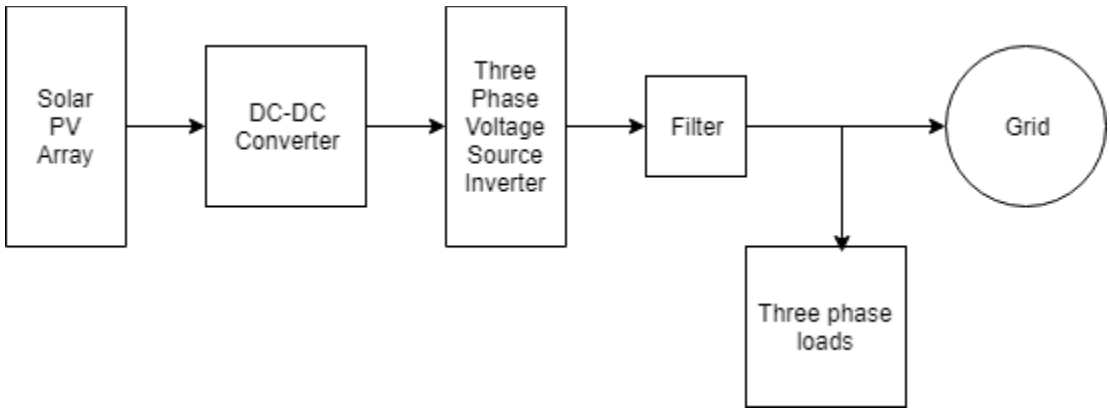

Fig. 5-1: General solar energy conversion system.

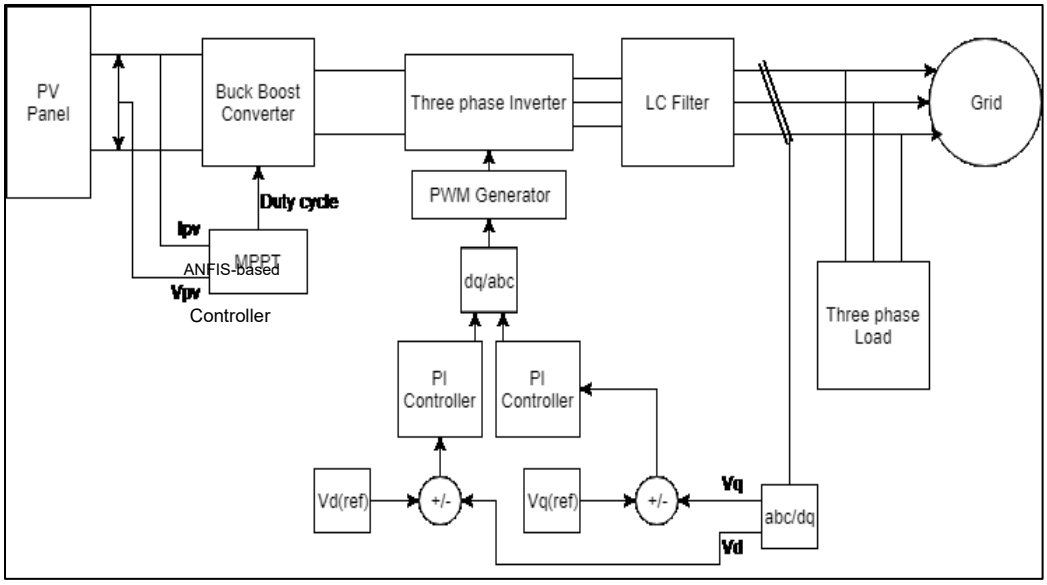

Fig. 5-2: Proposed solar energy conversion system.

Note that the MPPT block shown in Fig. 5-2 above is an adaptive neuro-fuzzy inference system that uses the proposed RLSE-PSO training algorithm to train the ANFIS and provide a superior maximum power point tracking response. Further, note that for real time implementation, instead of a LC filter, an inductor filter is used and the three phase resistive load is removed. Without the load, direct connection to the grid is achieved and all power from the inverter is fed to the grid.

To tune the PI controllers used for inverter control, the trial and error method is used to determine the four gains of the two controllers. The controllers were tuned until the desired response for voltage and current was obtained at the output of the inverter. With changing values of the proportional and integral gains of the d-axis component controller, the three phase output voltage and current can be made almost perfectly sinusoidal with the desired voltage level. The q-axis component controller has a very small effect on the response of the inverter. The desired output voltage of the converter is a three phase sinusoidal signal that has 30V peak phase voltage and a frequency of 60Hz.

A purely resistive three phase load is used for the proposed system in full scale system simulations (see 5.2 and 5.3). For the real time system simulations, the resistive load is removed. Depending on the resistance value of the load, the peak value of the current through the load will change according to I=30/R. Since the load is purely resistive, the power factor of the inverter

output is one. As such, the RMS output power of the inverter (power of the load) can be calculated as,

$$P_{out,AC} = V_{load,RMS} I_{load,RMS} \tag{113}$$

The PV output power is calculated as,

$$P_{PV,dc} = V_{PV} I_{PV} \tag{114}$$

The efficiency of the energy conversion system is defined as,

$$\eta = \frac{P_{out,AC}}{P_{in,dc}} \tag{115}$$

where, $P_{in,dc}=P_{PV,dc}$

Note that the duty cycle of the PWM signal applied to the gate of the switch affects the efficiency of the converter. When the duty cycle varies, the conduction losses and the switching losses of the diode and switch change. Since the losses change with a varying duty cycle, the efficiency of the converter must also change.

Assume there are no losses from the solar panel to the grid (i.e. 100% efficient converter and inverter). If the PV output power is equal to this RMS AC output power, the current from the inverter is fully consumed by the load and no current is fed to the grid. If the PV output power is less than the required AC RMS load power, all of the inverter output current will be consumed by the load and extra current will be supplied by the grid to the load. If the PV output power is greater than the required AC RMS load power, the load draws its required current and the remaining current is fed into the grid. The efficiency of the DC-AC conversion depends on the solar panel output power as well as the three phase load connected to the inverter output. The grid voltage is assumed to be fixed so a change in load changes the output current hence changing the output power. Further, the MPP voltage of the solar panel is fixed since the system is designed for 25°C. The irradiance level changes the MPP current outputted by the panel and hence changes the DC power supplied by the panel. Therefore, if the irradiance changes or the load changes, the efficiency of the energy conversion system also changes.

5.1 Recursive Least Squares Estimator Design

Chapter 6 discusses the training and response of the system. As such, the trial and error based design of the particle swarm optimization algorithm is discussed there. For the ANFIS design in this thesis, all input membership functions are chosen to be Gaussian functions. The input membership functions are:

$$\mu_{A1}(v_{pv}) = e^{-0.5\left(\frac{v_{pv}-c_1}{\sigma_1}\right)^2} \tag{116}$$

$$\mu_{A2}(v_{pv}) = e^{-0.5\left(\frac{v_{pv}-c_2}{\sigma_2}\right)^2} \tag{117}$$

$$\mu_{B1}(i_{pv}) = e^{-0.5\left(\frac{i_{pv}-c_3}{\sigma_3}\right)^2} \tag{118}$$

$$\mu_{B2}(i_{pv}) = e^{-0.5\left(\frac{i_{pv}-c_4}{\sigma_4}\right)^2} \tag{119}$$

where,

(c_i, σ_i), i=1, 2, 3, 4 are the nonlinear parameters

(p_i, q_i, r_i), i=1, 2, 3, 4 are the linear parameters

With four rules and two membership functions per input, there are eight nonlinear parameters and 12 linear parameters to optimize. As explained in the theory section, since online training is used, the recursive least squares estimator will be used to train the linear consequent parameters of the fuzzy system. For the designed system, the system output is:

$$x_{out} = \overline{W_1}F_1 + \overline{W_2}F_2 + \overline{W_3}F_3 + \overline{W_4}F_4$$

$$= \left(\overline{W_1}v_{pv}\right)p_1 + \left(\overline{W_1}i_{pv}\right)q_1 + \left(\overline{W_1}\right)r_1 + \left(\overline{W_2}v_{pv}\right)p_2 + \left(\overline{W_2}i_{pv}\right)q_2 + \left(\overline{W_2}\right)r_2 + \left(\overline{W_3}v_{pv}\right)p_3 + \left(\overline{W_3}i_{pv}\right)q_3 + \left(\overline{W_3}\right)r_3 + \left(\overline{W_4}v_{pv}\right)p_4 + \left(\overline{W_4}i_{pv}\right)q_4 + \left(\overline{W_4}\right)r_4 \tag{120}$$

In matrix form, the above equation can be represented as:

$$x_{out} = a\theta \tag{121}$$

where,

$$a = [\overline{W_1}v_{pv}, \overline{W_1}i_{pv}, \overline{W_1}, \overline{W_2}v_{pv}, \overline{W_2}i_{pv}, \overline{W_2}, \overline{W_3}v_{pv}, \overline{W_3}i_{pv}, \overline{W_3}, \overline{W_4}v_{pv}, \overline{W_4}i_{pv}, \overline{W_4}]$$

$$\theta = [p_1,\ q_1, r_1, p_2,\ q_2, r_2, p_3,\ q_3, r_3, p_4,\ q_4, r_4]^T$$

Thus, the above a and θ vectors are used in programming the RLSE to optimize the linear parameters for the designed ANFIS.

5.2 Full-scale Converter Design

This section provides a detailed discussion on the design of the full-scale buck boost converter used for simulation. The proposed MPPT algorithm is originally designed and tested through simulation on a full size grid connected system. The inverter peak output phase voltage is chosen to be 169.71V as this is the peak grid voltage. The input voltage of the converter (or solar panel maximum power point voltage) is chosen as 96V. The output voltage ripple is chosen as 1V and the inductor current ripple is chosen as 0.5A. The converter is designed as if it has a 10Ω load resistor. In practice, however, the converter is operating as an impedance matcher and is connected to the inverter. Therefore, the actual load is not 10Ω but the design still holds with the unknown load on the converter. The switching frequency of the converter is chosen as 50 kHz. The specifications of the design are summarized below:

Table 5-1: Full-scale converter specifications

Parameter	Value
V_s	96V
V_o	170V
R_L	10Ω
I_o	17A
Δi_L	0.5A
ΔV_o	1V
f_s	50 kHz

The duty cycle of the converter is,

$$D = \frac{V_o}{V_o + V_s} \tag{122}$$

$$= \frac{170}{96 + 170}$$

$$D = 0.64$$

This duty cycle is required to determine the value of the converter's inductor and capacitor as follows:

$$L \geq \frac{V_s D T_s}{\Delta i_L} \tag{123}$$

$$= \frac{(96)(0.64)\left(\frac{1}{50000}\right)}{0.5}$$

$$L \geq 2.46mH$$

$$c \geq \frac{I_o D T_s}{\Delta v_o} \tag{124}$$

$$= \frac{(17)(0.64)\left(\frac{1}{50000}\right)}{1}$$

$$c \geq 218\mu F$$

With these designed values, the inductor is chosen as 5mH and the capacitor is chosen as 400μF. Now with the new inductor and capacitor values, operation in continuous conduction mode (CCM) must be verified. For CCM of operation, $I_{OB} < I_O$. In this design, I_O is 17A.

$$I_{OB} = \frac{(1-D)^2 V_o T_s}{2L} \tag{125}$$

$$= \frac{(1 - 0.65)^2 (170)(\frac{1}{50000})}{2(5x10^{-3})}$$

$$I_{OB} = 0.042A$$

Here, $I_O > I_{OB}$ and hence, continuous conduction mode of operation is verified with the new inductor value.

5.3 Scaled-down Converter Design

This section provides a detailed discussion on the design of the buck boost converter used for the scaled down converter used for real time implementation. In order to lower the costs of the physical system, a scaled down model is designed to test the proposed MPPT algorithm. The inverter peak output phase voltage is chosen to be 30V. This inverter output voltage requires a dc inverter input voltage of 45V (the dc link voltage). Therefore, the converter must be designed to be able to have 45V as its output voltage. The input voltage of the converter (or solar panel maximum power point voltage) is chosen as 40V. The output voltage ripple is chosen as 0.5V (or 1.1%) and the inductor current ripple is chosen as 0.25A. The converter is designed as if it has a 10Ω load resistor. In practice, however, the converter is operating as an impedance matcher and is connected to the inverter. Therefore, the actual load is not 10Ω but the design still holds with the unknown load on the converter. The switching frequency of the converter is chosen as 50kHz. The solar emulator used for the real time implementation can generate a MPP voltage between 1 and 7 times the base voltage of 7.8V for a single cell and the MPP current between 1 and 45 times the base current of 95mA for the single cell. To increase the voltage, cells are placed in series and to increase the current, the cells are placed in parallel. Therefore, the emulator can output a maximum of 233.42W. The source voltage at the MPP is chosen to be 39V which corresponds to five cells placed in series and the source current at the MPP is chosen to be 2.09A which corresponds to 22 cells in parallel. The specifications of the design are summarized below:

Table 5-2: Real-time implementation converter specifications

Parameter	Value
V_s	39V
V_o	45V
R_L	10Ω
I_o	4.5A
Δi_L	0.25A
ΔV_o	0.5V
f_s	50 kHz

The duty cycle of the converter is,

$$D = \frac{V_o}{V_o + V_s} \tag{126}$$

$$= \frac{45}{45 + 39}$$

$$D = 0.54$$

This duty cycle is required to determine the value of the converter's inductor and capacitor as follows:

$$L \geq \frac{V_s D T_s}{\Delta i_L} \tag{127}$$

$$= \frac{(39)(0.54)\left(\frac{1}{50000}\right)}{0.25}$$

$$L \geq 1.7\ mH$$

$$c \geq \frac{I_o D T_s}{\Delta v_o} \tag{128}$$

$$= \frac{(4.5)(0.54)\left(\frac{1}{50000}\right)}{0.5}$$

$$c \geq 97.2\ \mu F$$

For real time implementation, the output capacitor has a rating of 100V therefore the duty cycle must be limited so the output voltage of the converter does not exceed this rating. Further, the minimum value of the duty cycle is also specified to prevent damage to the converter. The range of duty cycle is from 0.15 to 0.7. With a duty cycle of 0.15 and 0.7, the output voltage is 5.5V and 72.8V respectively. If the duty is allowed to be 0.8, the output voltage exceeds the capacitor rating. If the duty is allowed to be 0.75, the output voltage is 93.6V which is close to the maximum rating of the capacitor so for safety reasons, a maximum duty cycle of 0.7 is chosen.

With a duty of 0.15, the designed inductor and capacitor values will be less than the above values so the converter will still operate as designed for a duty of 0.15. For a duty cycle of 0.7, the inductor value must be greater than 2.2 mH and the capacitor must be greater than 126 µF. Therefore, the inductor and capacitor values must be chosen according to the above restrictions in order for the converter to operate as designed with a duty cycle of 0.7.

With these designed values, the inductor is chosen as 2.5mH and the capacitor is chosen as 470μF. Now with the new inductor and capacitor values, operation in continuous conduction mode (CCM) must be verified. For CCM of operation, $I_{OB} < I_O$. In this design, I_O is 4.5A.

$$I_{OB} = \frac{(1-D)^2 V_o T_s}{2L} \tag{129}$$

$$= \frac{(1-0.54)^2(45)(\frac{1}{50000})}{2(2.5x10^{-3})}$$

$$I_{OB} = 0.038A$$

Here, $I_O > I_{OB}$ and hence, continuous conduction mode of operation is verified with the designed inductor value.

5.4 LC Filter Design

This section discusses the design of the low pass LC filter used to smooth out the inverter output voltage and current. The filter further removes higher order harmonics and mostly retains only the fundamental component. Specifically, the third harmonic is the most important harmonic to filter out, therefore the filter cut off frequency is chosen as 60Hz.

To design the filter, the cut off frequency and a standard inductor value with the required current rating available for purchase are chosen. From these values, the required capacitor value is then calculated. For this design, the inductor value is chosen to be the same as the converter's inductor with a value of 2.5mH. The capacitor is calculated as,

$$c = \frac{1}{L}\left(\frac{1}{2\pi f_c}\right)^2 \tag{130}$$

$$= \frac{1}{2.5x10^{-3}}\left(\frac{1}{2\pi(60)}\right)^2$$

$$c = 2.814mF$$

This designed capacitance value is not available for purchase, therefore, a closely valued capacitor that is available for purchase is chosen as 2.7mF. Now, the cut off frequency is

changed since a different capacitor value is used. Therefore, the new cut off frequency must be calculated to verify the filter will still meet the design requirements.

$$f_c = \frac{1}{2\pi\sqrt{LC}} \tag{131}$$
$$= \frac{1}{2\pi\sqrt{(2.5x10^{-3})(2.7x10^{-3})}}$$
$$f_c = 61.26Hz$$

Therefore, with the new capacitor and the chosen inductor, the cut off frequency is only increased by 1.26Hz and hence will still remove any frequencies above the fundamental three phase frequency of 60Hz. Further, the frequency response of the filter can be obtained in MATLAB which shows the filter's pass band and rejection (or attenuation) region. Here, the transfer function of the filter is:

$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{LCs^2+1} \tag{132}$$
$$\frac{V_{out}(s)}{V_{in}(s)} = \frac{1}{6.75x10^{-6}s^2 + 1}$$

From this transfer function, the bode function in MATLAB is used to generate the frequency response as shown in Fig. 5-3.

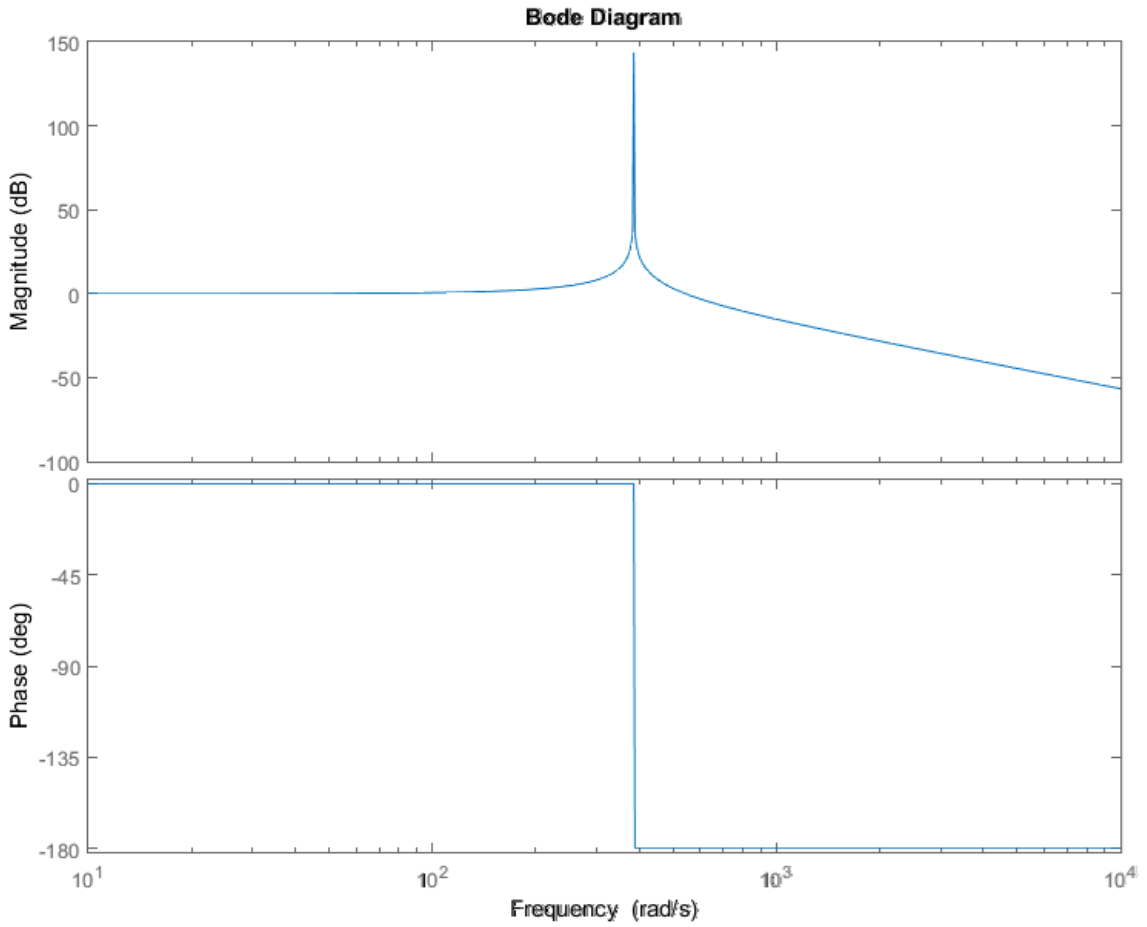

Fig. 5-3: LC filter frequency response.

It is clearly illustrated that the filter has its peak at 385 rad/s or 61.26 Hz hence proving the design of the filter. With this filter design, all higher order harmonics are attenuated and filtered out of the three phase inverter output signal with only the fundamental component remaining and having its full magnitude. It should be noted that a higher cut-off frequency could be designed which would ease the requirements on the inductor and capacitor values.

Further, for the purposes of real-time experimentation, the LC filter is not used to filter the output of the inverter. Instead, a series inductor will be used for each phase to filter out higher order harmonics from the inverter output. This decreases the cost of the filter as AC safety capacitors are expensive when creating a high capacitance value. The inductor filter is sufficient for filtering as it provides enough attenuation for higher frequency harmonic components. Since m_f is large, the harmonics are located far away from the fundamental frequency and filtering

them is not a significant issue. A larger inductor value should create a smoother output signal resulting in less harmonic distortion with a larger resistance and voltage drop across it, when compared to an inductor with a smaller inductance.

Chapter 6

Simulation of the Proposed ANFIS Based MPPT Control of PV Solar System

Provided in this chapter are the results from simulating both of the designed systems (full scale system and scaled down system) in the MATLAB/Simulink environment. The Simscape toolbox, specifically SimElectronics, in Simulink is used to simulate the designed power electronic system.

6.1 Full-scale System Simulation

Before the proposed system is trained, training data is gathered by operating the system with the P&O MPPT algorithm. Since the P&O algorithm only affects the operating point of the solar panel, and does not significantly affect the dc link or inverter output, only the solar panel output will be provided for operation with P&O MPPT. Note that the results provided in this section are obtained by simulating the full-scale system. The scaled down system has a similar response and waveform shapes as the full size system discussed in the next section. Real time training and real time results of the scaled down system are provided in the next chapter. For this simulation, the temperature is assumed to be constant at 25°C while the irradiance is stepped among various values. Further, the LG Electronics LG300N1C-G3 panel is used in Simulink. The specifications of the PV panel are shown in Fig. 6-1. Note that three panels are placed in series to achieve a MPP voltage of 96V. The PV curves of this panel are shown in Fig. 6-2. Fig. 6-3 shows the power output of the solar panel for the P&O MPPT. As shown in Fig. 6-3, the perturb and observe MPPT algorithm suffers from oscillations around the maximum power point. At an irradiance of 1000W/m^2, the maximum possible power output of the solar panel is around 908W. With the P&O MPPT algorithm operating the buck boost converter, the average solar panel output power is around 870W. Thus, there is a power loss of around 40W. This power loss shows a disadvantage of the P&O algorithm as discussed previously. The oscillations at the MPP cause

a decrease in available power and a lower system efficiency results. For simulation purposes, the inverter is connected to a 10 ohm three phase wye connected resistive load as well as the grid.

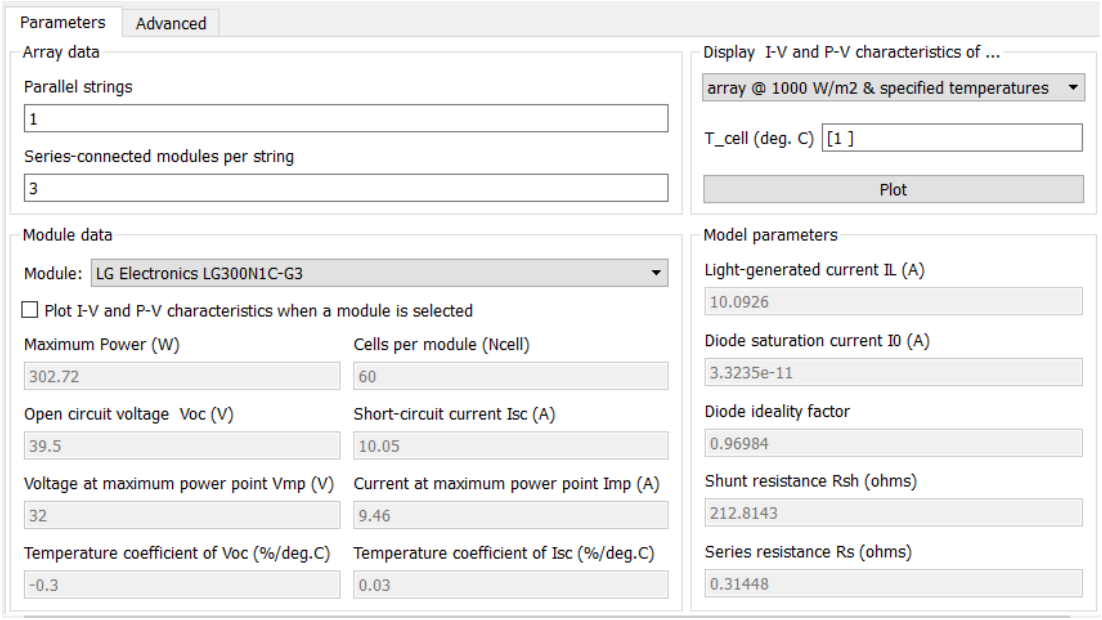

Fig. 6-1: Solar panel specs.

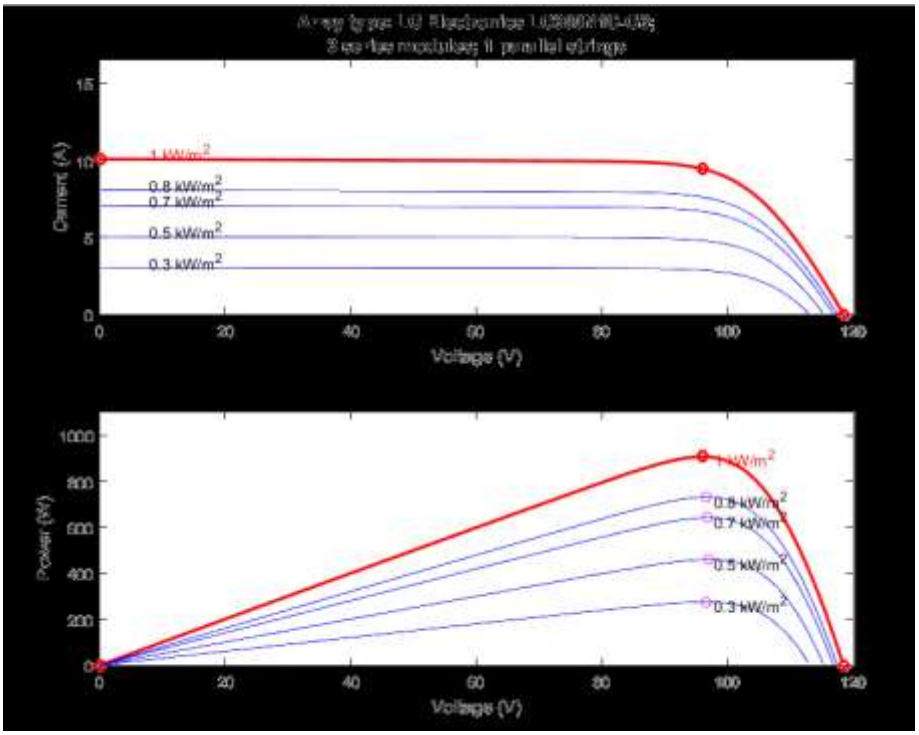

Fig. 6-2: PV curves for LG panel.

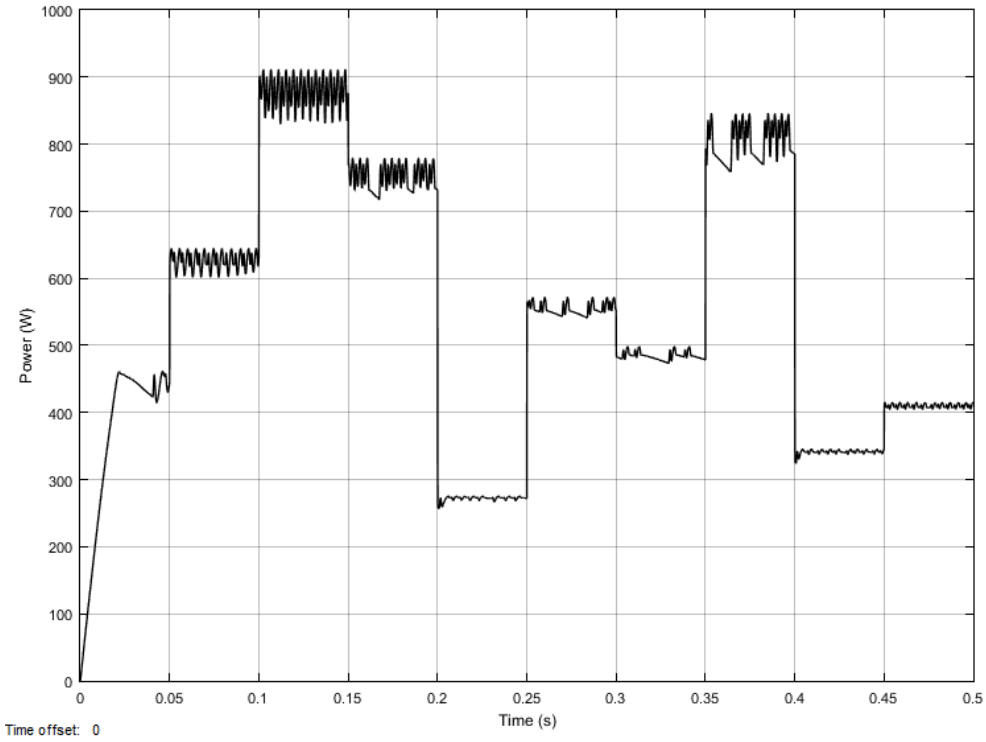

Fig. 6-3: Solar panel output power using P&O MPPT.

From gathering system data while running the P&O algorithm, the proposed ANFIS controller can be trained and a better response obtained. Fig. 6-4 shows the testing data (desired output) and the output of the ANFIS after the training is completed. Note that the system data is highly nonlinear but the ANFIS is able to find the pattern in the system data and is effectively trained using the proposed RLSE-PSO training algorithm.

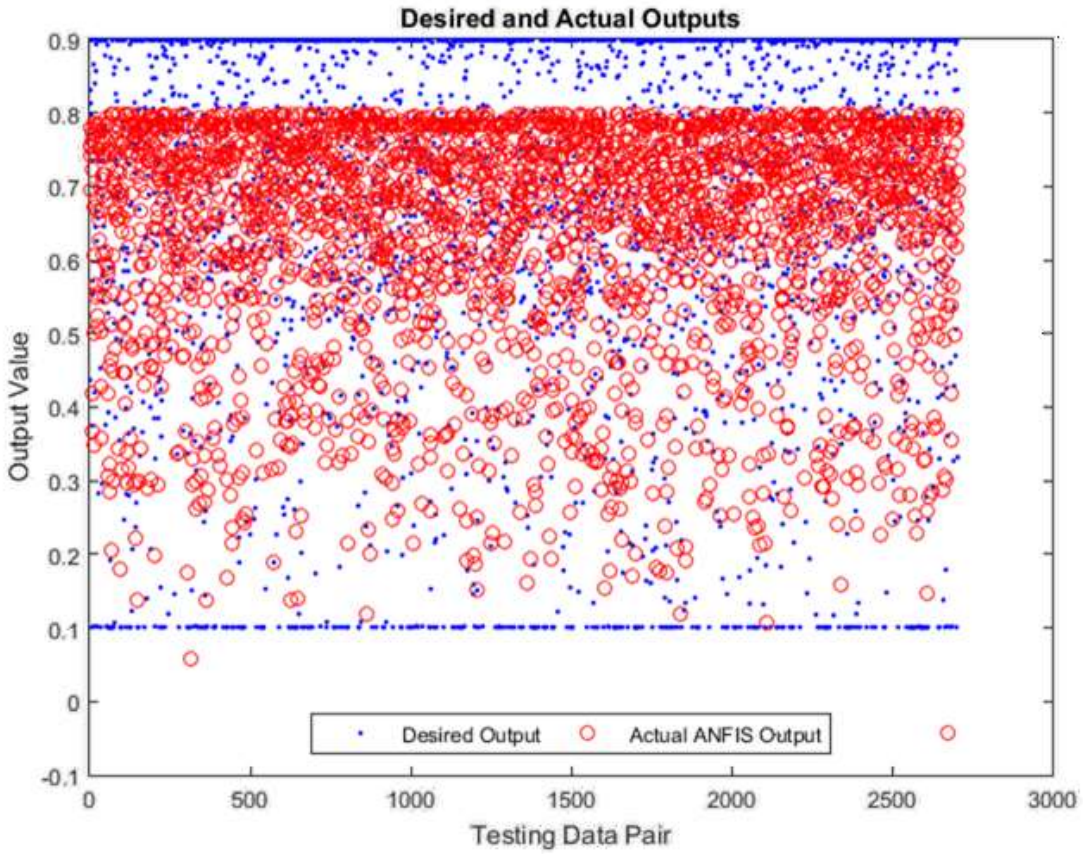

Fig. 6-4: Testing data and ANFIS output.

The ANFIS calculates a duty cycle that is applied to the switch of the converter in order to operate the solar panel at its maximum power point. This operation allows maximum possible power to be transferred to the DC link at all times. For training, the root mean square error (RMSE) tolerance is set at 20% and the ANFIS is trained for 85 epochs using approximately 3000 data points per epoch. Training was completed for more than 85 epochs but no change in the error function was observed. Since the error converges in 85 epochs, no more epochs are required for training purposes. Therefore, training is completed again and stopped once 85 epochs or the desired training error is reached. Fig. 6-5 shows a plot of the training error versus epoch number. Although the training error is around 0.203, the output of the ANFIS matches closely to the training data.

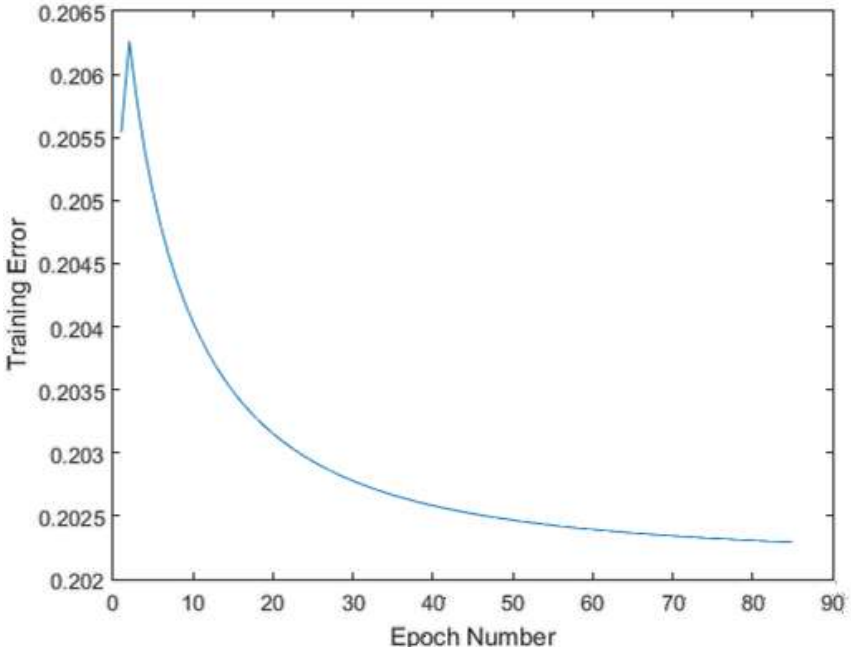

Fig. 6-5: Training error.

The resulting membership functions after system training for PV voltage and PV current are shown in Fig. 6-6. As shown in Fig. 6-6, the membership functions cover the entire range of operation and hence the training is verified. Fig. 6-7 shows the PV output voltage and current for the proposed system using the ANFIS as the MPPT controller with varying irradiances. An irradiance change is simulated every 50 ms with both increases and decreases in irradiance. Fig. 6-8 shows the output power of the solar panel for the proposed system. Table 6-1 compares the actual power output of the panel and theoretical power output (obtained from the PV curves) at various irradiances for the proposed system. It is shown that the ANFIS is very effective at forcing the solar panel to operate at its MPP with a fast response and almost no oscillations at the MPP regardless of irradiance conditions. As the proposed system has almost no oscillations at the MPP, almost no power is lost and the system efficiency is increased as compared to the conventional P&O based system. It is shown that when the output current of the panel changes due to varying irradiance conditions, the output voltage of the panel adjusts to achieve maximum power output. The proposed ANFIS based MPPT algorithm outperforms conventional P&O based MPPT by achieving a faster response and fewer oscillations at each MPP. These results validate the proposed algorithm and show that the proposed algorithm has less power loss due to oscillations at the MPP. Hence, the proposed system is able to transfer more power to the dc link

resulting in a more efficient system as compared to the system using conventional P&O MPPT. Note that the initial membership functions do not matter as the proposed algorithm determines the best fitting membership functions regardless of starting position. The initial membership functions are randomly created at the beginning of the training.

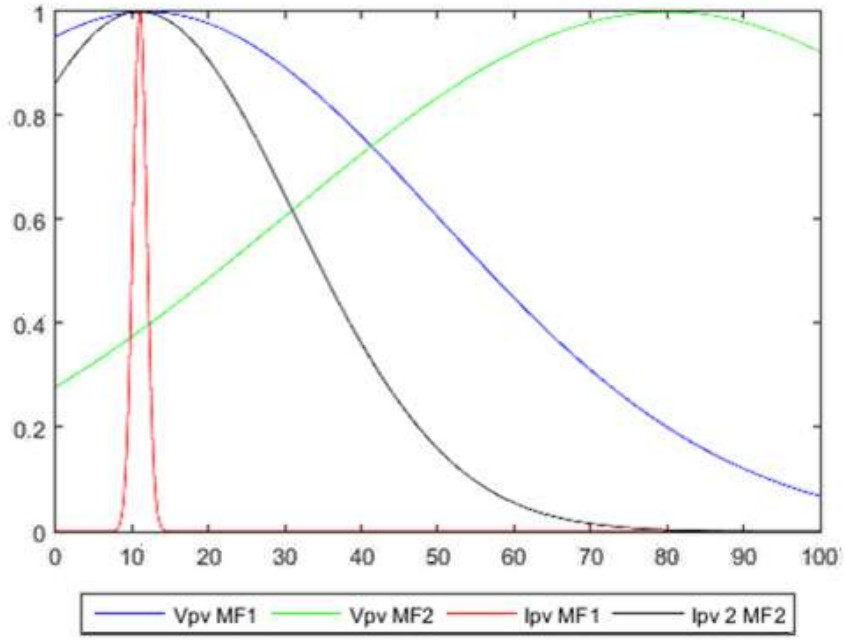

Fig. 6-6: Input membership functions after training.

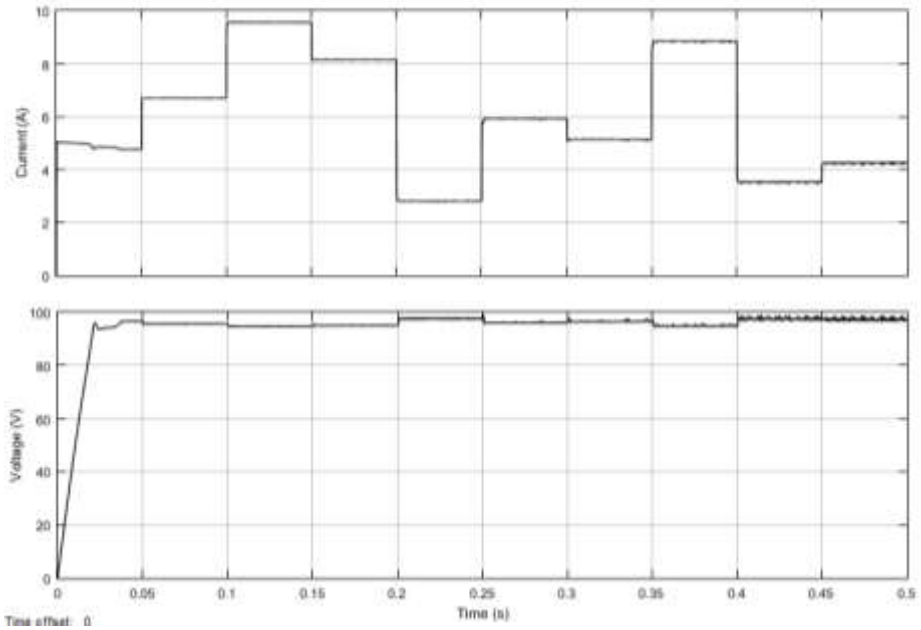

Fig. 6-7: Solar panel output voltage and current for the proposed system.

The irradiance is changed every 50ms with the irradiance vector of [500, 700, 1000, 850, 300, 620, 540, 925, 375, 450] W/m^2.

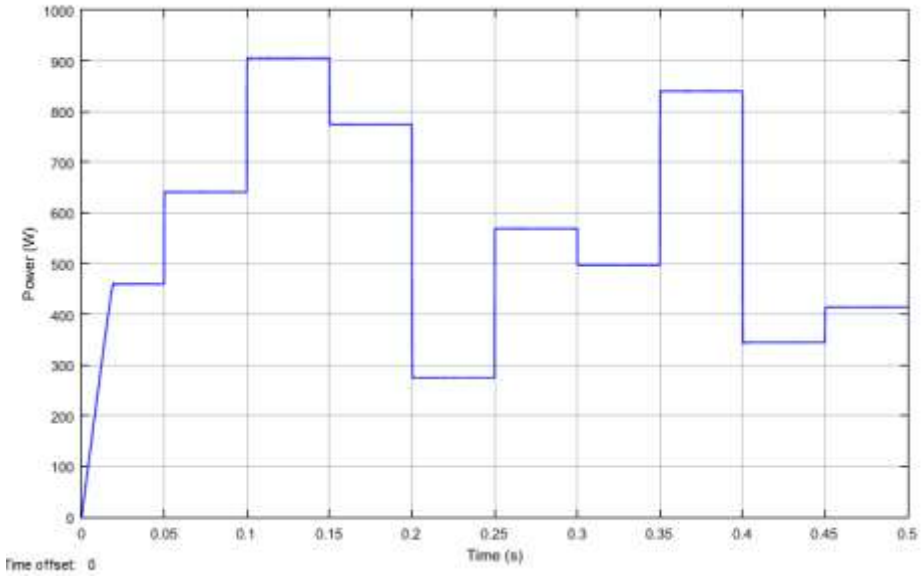

Fig. 6-8: Solar panel output power for the proposed system.

Even if the dc link voltage varies in the simulation, the inverter is able to achieve a constant output once steady state is reached. Inverter voltage control is implemented in order to achieve effective grid connection; therefore, the current will change with changing dc link voltages. The following figures are obtained for the proposed system running the proposed ANFIS MPPT controller. Fig. 6-9 shows the PWM control signals that are applied to inverter switches to control the output voltage and achieve a three phase output with abc sequence and a frequency of 60Hz. Fig. 6-10 and Fig. 6-11 show the three phase voltage and current respectively at the output of the inverter for the proposed system. Fig. 6-12 shows the load three phase load current at the output of the inverter. After some time, the inverter output current reaches steady state. It is shown that the load current, inverter output current and output voltage are almost perfectly sinusoidal with abc sequence and a frequency of 60Hz. The total harmonic distortion for the voltage and currents is less than 1%. Fig. 6-13 and Fig. 6-14 show the THD plot for the voltage and current of phase ‘a’ respectively. Similar results for the THD are obtained for ‘b’ and ‘c’ phases. The low THD proves that the designed filter is effective at removing unwanted harmonics and retains only the fundamental component of the waveforms. Thus, effective grid connection is possible.

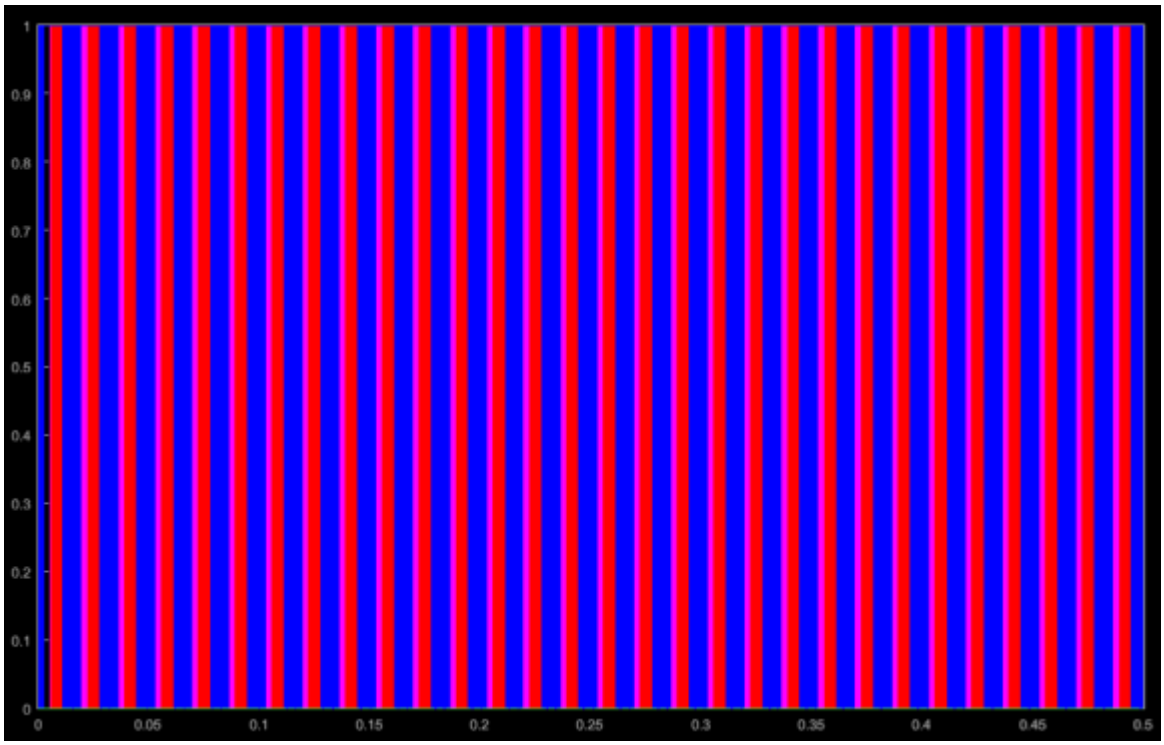

Fig. 6-9: Inverter PWM switching signals.

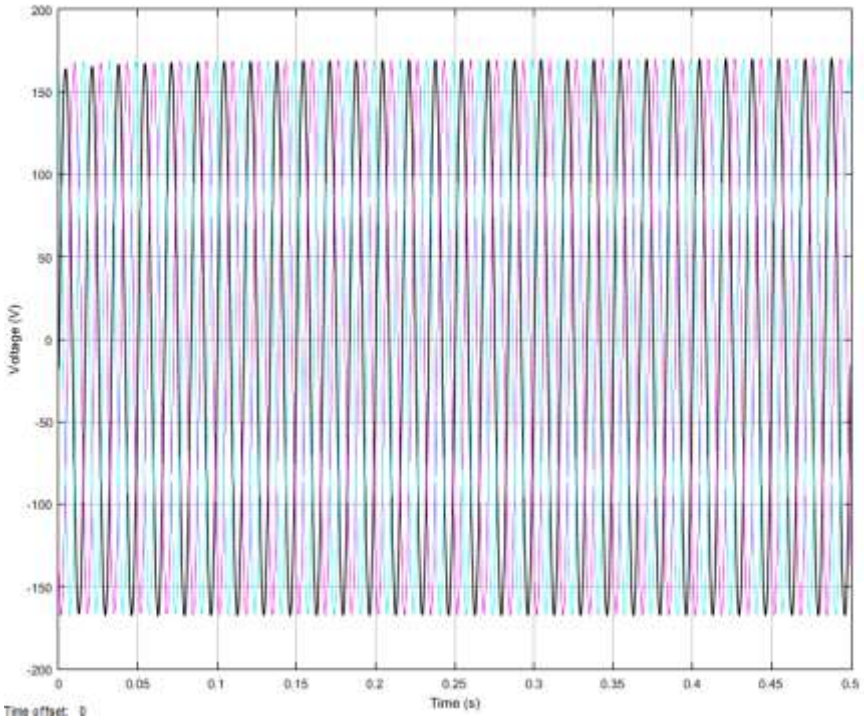

Fig. 6-10: Three phase output voltage at inverter end.

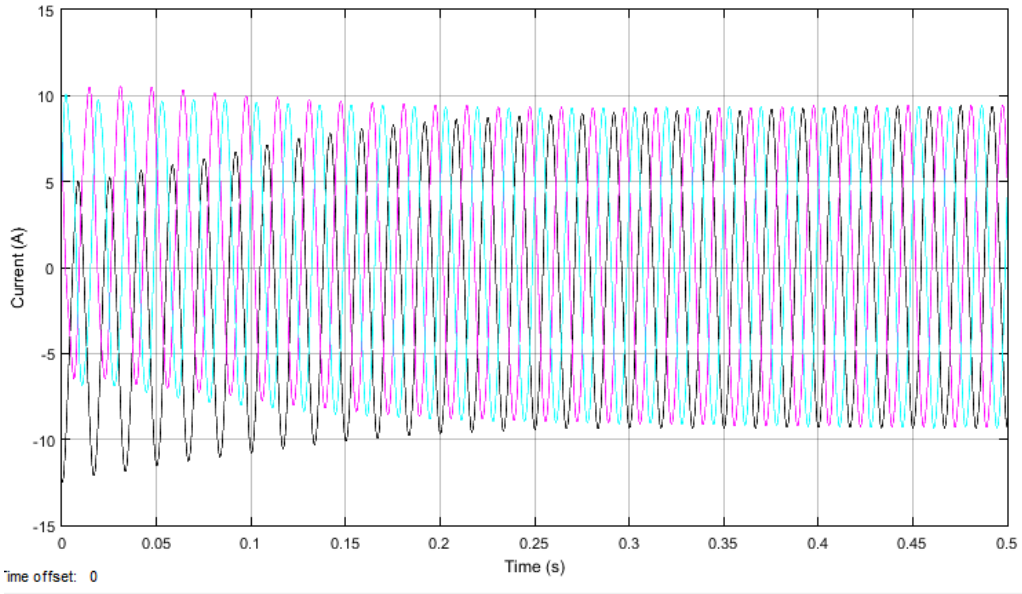

Fig. 6-11: Inverter output current.

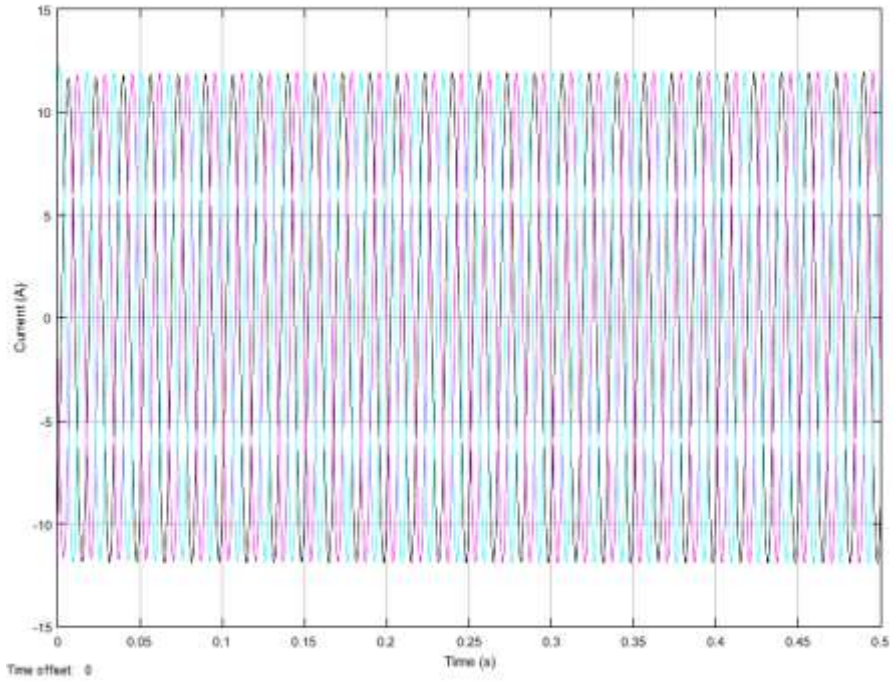

Fig. 6-12: Load current.

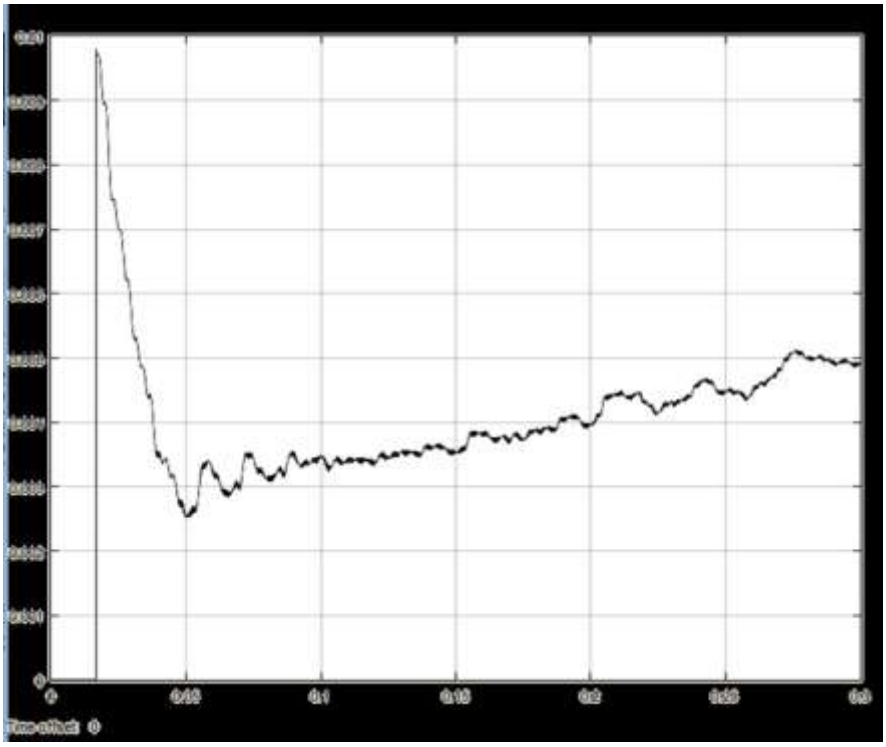

Fig. 6-13: Phase ‘a’ output voltage total harmonic distortion.

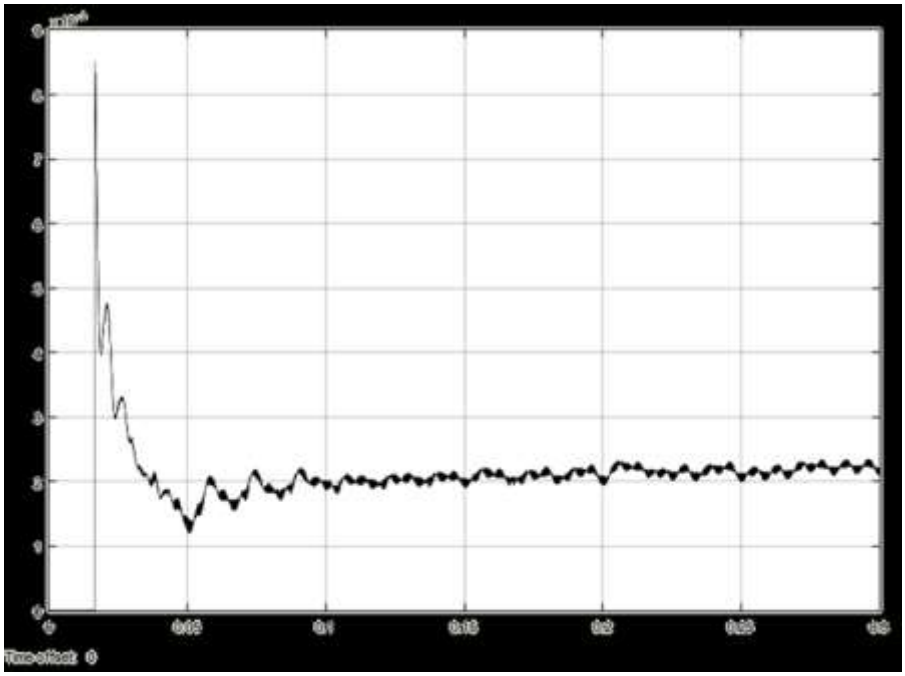

Fig. 6-14: Phase ‘a’ output current total harmonic distortion.

Simulink's built in FFT Analysis tool in the powergui block is used to perform the FFT on both the inverter output voltage and the inverter output current. The results of the FFT are shown below in a list form. It is proven that the inverter and filter are able to effectively generate a three phase output with mainly the first (fundamental) harmonic in the signal. All other harmonics are less than 1 percent of the signal for both voltage and current. It is shown that for both the output voltage and current, there is no dc offset and the third harmonic component is not present in the signals as its magnitude relative to the fundmental is 0.02%. The second harmonic has the largest magnitude in the signals but is only 0.63% of the fudamental magnitude. Note that the FFT results for the voltage and current are the same since the current flowing through the load is defined by the voltage across it.

Table 6-1: FFT results for phase 'a' phase voltage

0 Hz	(DC):	0.01%	270.0°
60 Hz	(Fnd):	100.00%	-1.6°
120 Hz	(h2):	0.63%	192.8°
180 Hz	(h3):	0.02%	-5.8°
240 Hz	(h4):	0.22%	107.7°
300 Hz	(h5):	0.05%	-4.1°
360 Hz	(h6):	0.01%	14.0°
420 Hz	(h7):	0.01%	-44.3°
480 Hz	(h8):	0.01%	61.6°
540 Hz	(h9):	0.00%	7.6°
600 Hz	(h10):	0.00%	35.3°
660 Hz	(h11):	0.00%	0.1°
720 Hz	(h12):	0.00%	0.5°
780 Hz	(h13):	0.00%	2.4°
840 Hz	(h14):	0.00%	6.8°
900 Hz	(h15):	0.00%	3.0°
960 Hz	(h16):	0.00%	1.4°

Table 6-2: FFT results for phase 'a' current

0 Hz	(DC):	0.01%	270.0°
60 Hz	(Fnd):	100.00%	-1.6°
120 Hz	(h2):	0.63%	192.8°
180 Hz	(h3):	0.02%	-5.8°
240 Hz	(h4):	0.22%	107.7°
300 Hz	(h5):	0.05%	-4.1°
360 Hz	(h6):	0.01%	14.0°
420 Hz	(h7):	0.01%	-44.3°
480 Hz	(h8):	0.01%	61.6°
540 Hz	(h9):	0.00%	7.6°
600 Hz	(h10):	0.00%	35.3°
660 Hz	(h11):	0.00%	0.1°
720 Hz	(h12):	0.00%	0.5°
780 Hz	(h13):	0.00%	2.4°
840 Hz	(h14):	0.00%	6.8°
900 Hz	(h15):	0.00%	3.0°
960 Hz	(h16):	0.00%	1.4°

The efficiency of the converter for varying irradiances (and hence varying duty cycles) with is found in Simulink. The buck boost converter is simulated with the proposed ANFIS-based

MPPT controller and connected to a resistive load. Note that the inverter is not connected for this efficiency simulation as only the efficiency of the buck boost converter is considered. A plot of the efficiency under varying irradiance condition is shown in Fig. 6-15. Note that the spikes in efficiency every 0.05 seconds are due to the transients generated when irradiance levels change. The converter efficiency is found to be 0.97 or 97%.

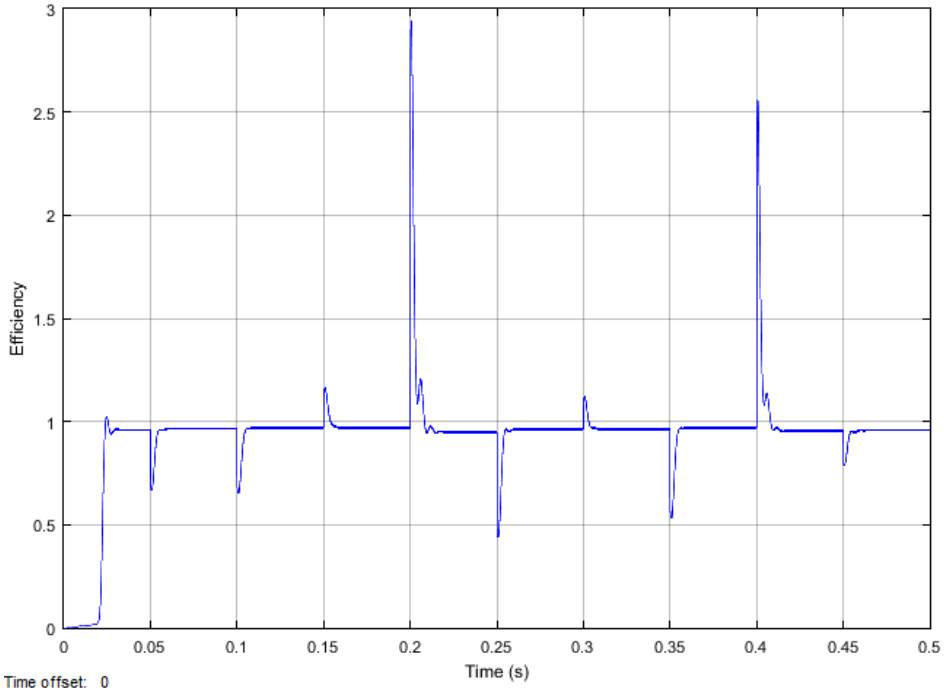

Fig. 6-15: Converter efficiency.

To further verify the effectiveness of the proposed RLSE-PSO training algorithm, the system reponse when trained with the proposed algorithm is compared to the system response when trained with the conventional hybrid training algorithm as introduced by Jang [1]. The same ANFIS structure is trained in MATLAB using the neuroFuzzyDesigner. The error tolerance and total epochs are set to the same value as when the ANFIS was trained with the proposed algorithm. The phase 'a' voltage THD for the proposed system is around 0.005 as shown in Fig. 6-13 above. The phase 'a' current THD for the proposed system is around 0.002 as shown in Fig. 6-14 above. The system trained with the conventional hybrid algorithm [1] has a higher THD for both of the inverter's output voltage and current. The THD for both voltage and current is around 0.008. As such, the proposed system generates a three phase output that is closer to a pure sinusoidal signal than the conventional system trained with GD. Fig. 6-16 below

shows the output power of the solar panel for the system trained with the conventional hybrid algorithm.

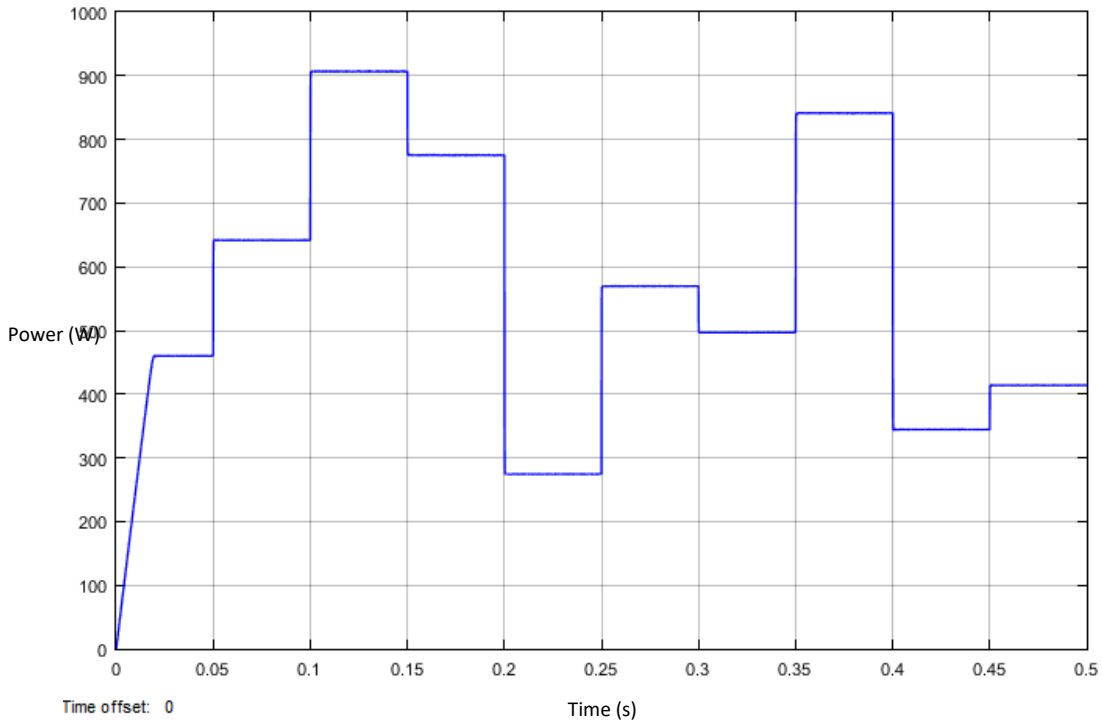

Fig. 6-16: Solar panel output power for the system trained with gradient descent algorithm.

Fig. 6-8 and Fig. 6-16 show that both the proposed system and the system trained with gradient descent have similar responses with little to no oscillations at the maximum power point. Both systems are able to reach the maximum power point quickly and both do not deviate from that operating point unless the conditions are changes. Further, Table 6-3 proves that the proposed ANFIS controller with buck boost converter is able to effectively force the solar panel to operate at the theoretical maximum power point hence transferring maximum power to the dc link at all times regardless of weather conditions. It is further shown that for all irradiances, the proposed system is able to extract more power from the solar panel than the conventional system with its ANFIS trained using the gradient descent algorithm. Also, the responses of both the proposed system and the conventional GD-based one are the same. When a change in irradiance is implemented (either an increase or decrease), both of the ANFIS are able to force the PV panel to reach the MPP in the same amount of time with the same response. Therefore, no benefit relating to tracking speed is seen from either training algorithm since the training algorithms are

applied to the same ANFIS. With the designed structure of the ANFIS, the controller can achieve PV operation at the MPP 0.5ms after an irradiance change is implemented. If the structure of the ANFIS is changed, the tracking speed may change. The only benefit of the gradient descent-based training is that the training is faster than the proposed algorithm. As a result, there is a trade-off between training speed and power extraction. If the slightly slower training time can be accepted, then the proposed system should be used as it has a better system response. The training is completed on an offline system however, so the extra training time should not be an issue for practical purposes. Therefore, it is proven that the proposed training algorithm is more effective than the conventional algorithm and hence a more efficient system with a higher power extraction results. Hence, the proposed system is more cost effective than both the system trained with GD and the system implementing the P&O MPPT controller. This cost effectiveness and increase in efficiency proves the proposed system's viability for practical industrial uses as compared to other MPPT methods. It is important to note that the proposed RLSE-PSO training algorithm can be used for any type of system in which data representing the system can be obtained.

Table 6-3: Comparison of PV output power for the proposed and conventional system

Irradiance (W/m^2)	Theoretical Power (W)	Proposed System Actual Power (W)	Conventional System Actual Power (W)
300	275	275	274.5
450	414	414	413.8
500	460	460	459.8
700	642	641.5	641.4
850	776	775	775
1000	908	907	906.8

6.2 Real-time System Simulation

This subsection provides the MATLAB/Simulink results for the simulation of the scaled down system used for real time implementation. Note that for this simulation, the three phase resistive load is not included and the filter is an inductor filter not a LC filter. The reasoning for removing the resistive load is that when performing the real time experiments, the attempted direct grid connection with full power transfer from the inverter to the grid can be achieved since no additional load will be present. Therefore, it is desirable to simulate the same system that will be used in practical implementation. The solar panel maximum power point voltage is 39V for all irradiances and has a power rating of 80W at an irradiance level of $1000W/m^2$. The amplitude of the three phase voltages at output of the inverter is set to 32V while the grid has amplitude of 30V.

As the results of the real time system simulation are very similar to the full scale system, a brief discussion of the simulation results will be provided as an in depth discussion can be found in section 6.1. The resulting membership functions after training the ANFIS with the P&O based system data are shown in Fig. 6-17. These membership functions along with the trained consequent parameters effectively force the solar panel to operate at its MPP for varying irradiance levels as shown by the solar panel output power in Fig. 6-18. Fig. 6-19 provides the three phase inverter output voltages (phase voltages). Fig. 6-20 displays the output current of the

inverter that is fed into the grid. From these results, effective grid connection is possible while operating the PV panel at its MPP. It is also determined that both of the voltage and current at the inverter end have a low total harmonic distortion hence verifying the inverter control scheme and filter operation. The Fast Fourier Transform (FFT) applied to the inverter output current is used to determine the total harmonic distortion of the signal. For the purposes of this discussion, only phase ‘a’ is considered, however, similar results are obtained for phases ‘b’ and ‘c’. From the FFT completed with the powergui module tools in Simulink, the THD for phase ‘a’ of the inverter’s output current is 2.86%. The results of the FFT are shown in Fig. 6-21.

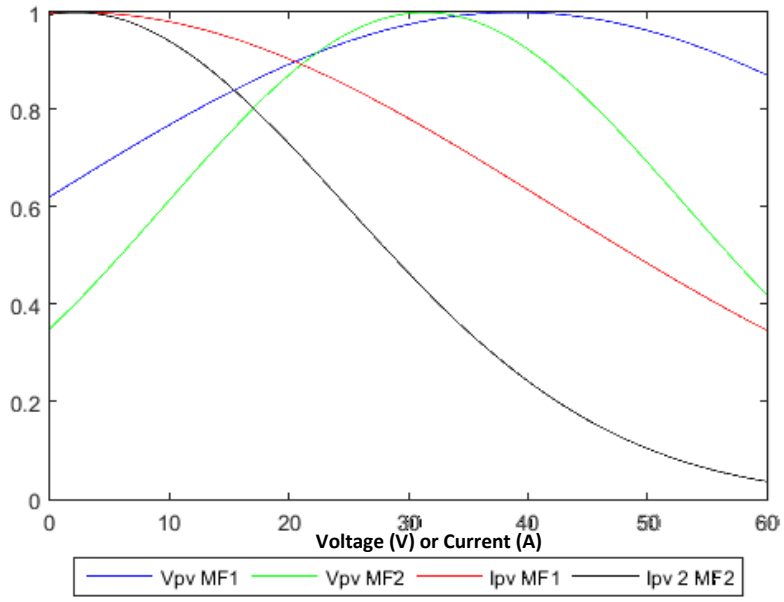

Fig. 6-17: Scaled down system membership functions.

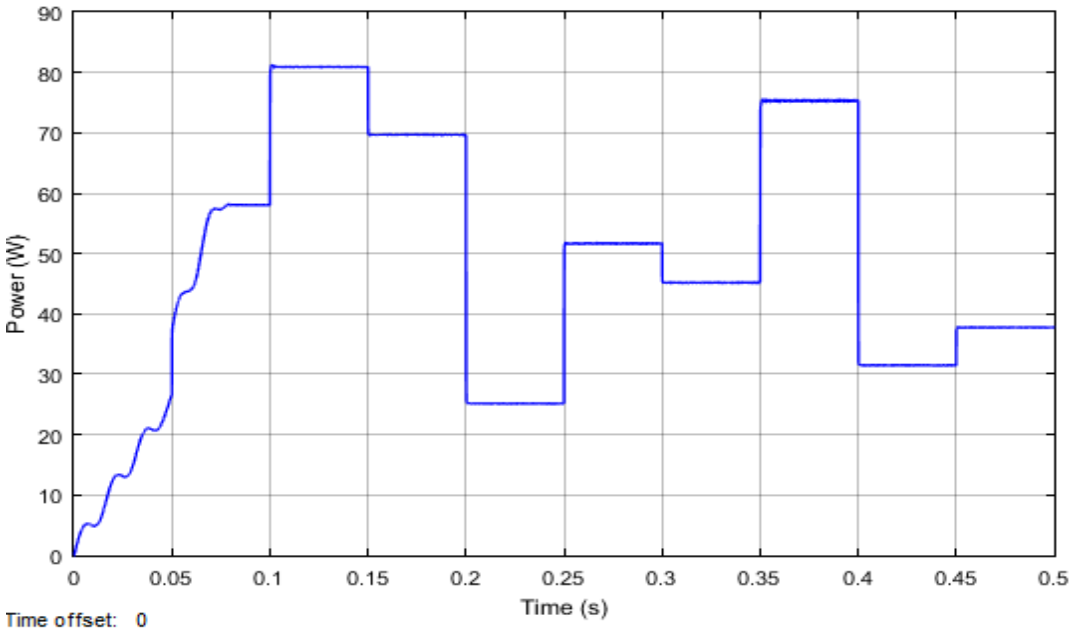

Fig. 6-18: Scaled down system PV output power.

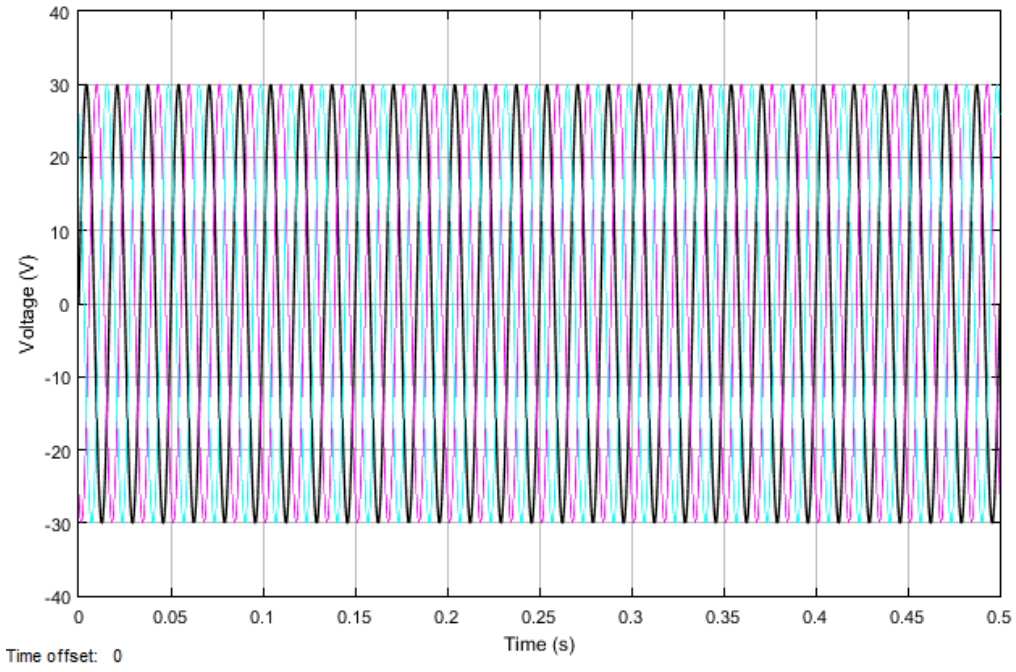

Fig. 6-19: Scaled down system phase voltages at the inverter end.

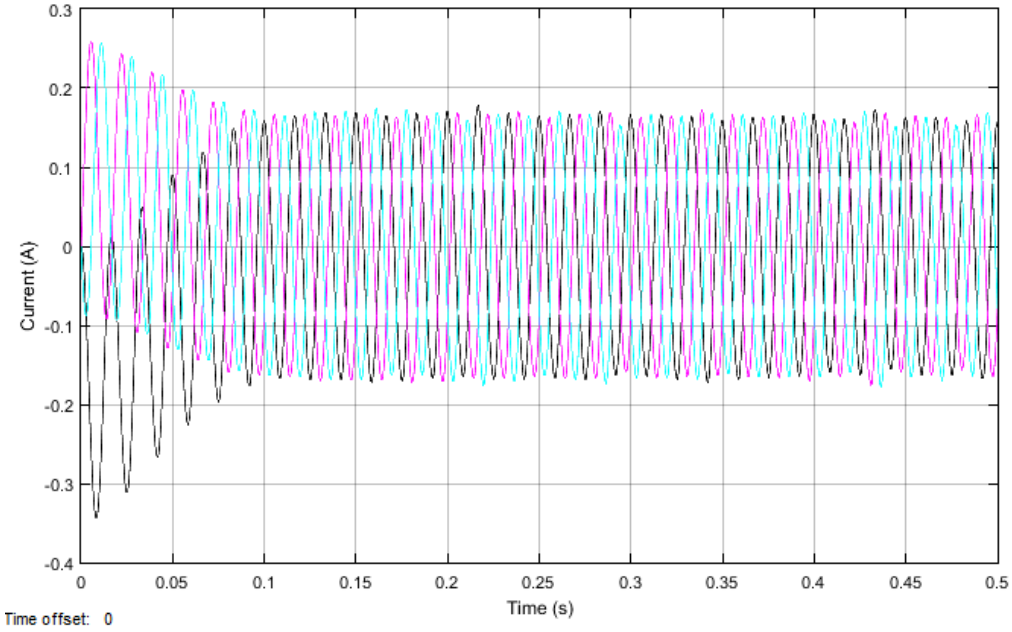

Fig. 6-20: Scaled down system inverter output current.

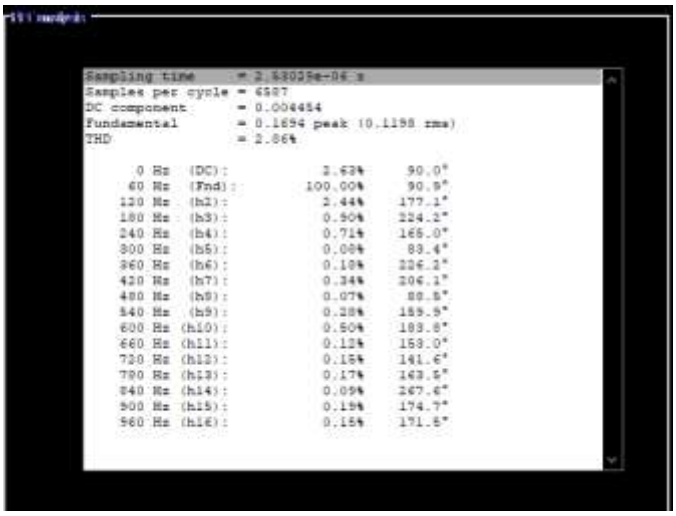

Fig. 6-21: Scaled down system FFT analysis for phase ‘a’ output current.

Considering only the solar array and buck boost converter with a resistive load, the following results are obtained. Fig. 6-22 shows the training error using P&O training data and Fig. 6-23 shows the PV output power with a 100Ω load connected to the output of the buck boost converter. It is clearly illustrated that the proposed training algorithm and proposed ANFIS-based MPPT controller is able to effectively force the solar panel to operate at its maximum

power point for a wide range of irradiance conditions. These results verify the proposed RLSE-PSO training algorithm as well as the designed ANFIS-based MPPT controller. These results are provided since the same system configuration is used for real-time implementation. The Simulink block diagram for this simulation can be found in Appendix B.

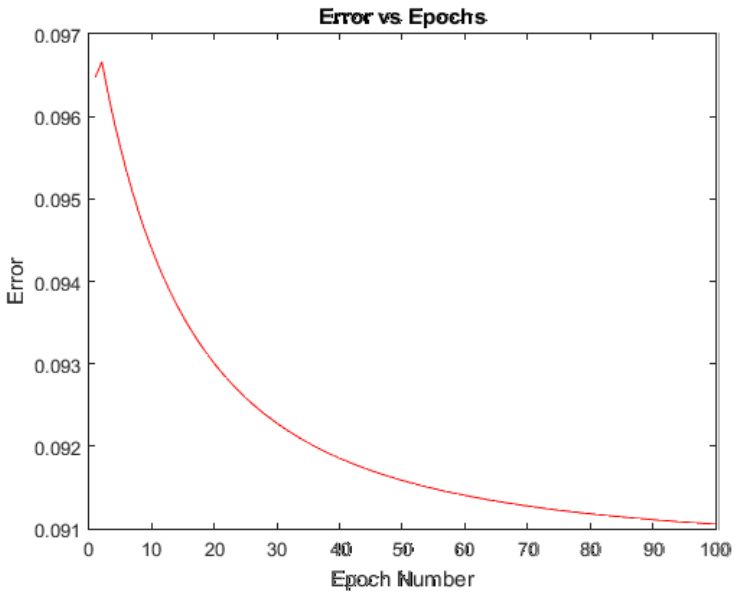

Fig. 6-22: Training error for the system with 100Ω DC load.

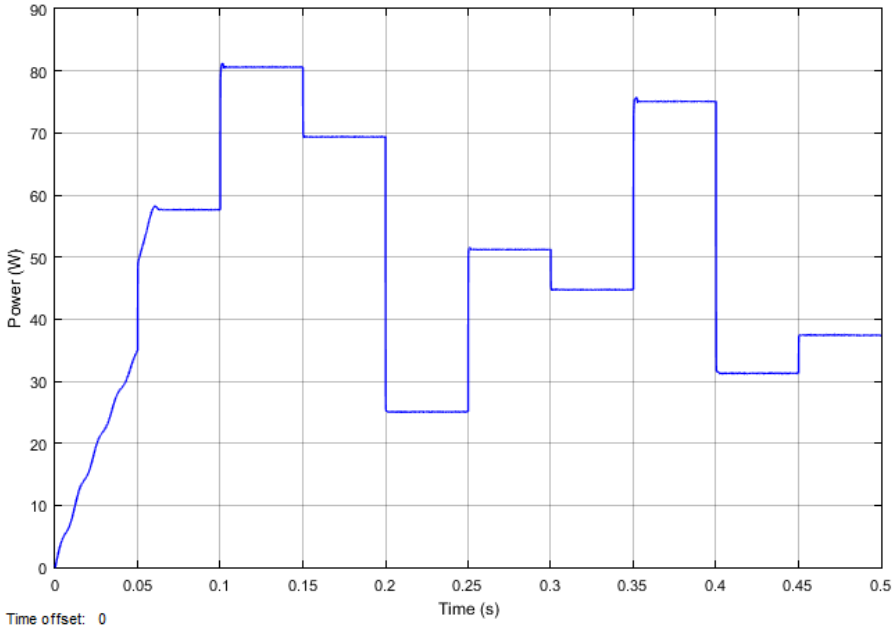

Fig. 6-23: PV output power for the system with 100Ω DC load.

Chapter 7

Real Time Implementation of the Proposed ANFIS Based MPPT Control of PV Solar System

The chapter provides the results obtained from physically implementing the designed system to prove the validity of the proposed maximum power point tracking algorithm. For the purposes of real time implementation, in order to reduce the cost of the prototype, the converter was redesigned to accommodate a lower input voltage, output voltage and output current. It is important to note that the ANFIS for real time implementation is trained using real time system data with the P&O maximum power point tracking algorithm.

For the real time implementation the dSPACE DS1104 board is used for system integration with Matlab/Simulink. Since it is not feasible to implement a real solar panel in the lab, the LabVolt Four-Quadrant Dynamometer (LabVolt Series 8930-20) is used. A solar emulator (Lab-Volt Material no. 8968-60 DIDACTIC TRAINING SOFTWARE Solar Panel Emulator) is implemented with the dynamometer so the characteristics of a solar panel can be obtained and used for system testing. Voltage sensors consisting of resistor configurations are used to scale down the solar panel emulator output voltage to a suitable range for the dSPACE board. Operational amplifiers connected as voltage followers are used to remove the loading effect on the measured voltage due to connection with the dSPACE board. Further, a Hall Effect current sensor is used to measure the output current of the solar panel emulator for MPPT algorithm operation. To reduce the number of power supplies needed, the 5V source voltage required to power the current sensor is generated with a 5V voltage regulator.

The PWM pulses are generated in Simulink and outputted to the converter through the dSPACE board. In order for the PWM pulses to switch the MOSFET on, a driver circuit is required. This driver circuit is created using the FOD3180 optoisolator/driver chip. Opto-isolation is required to protect the low power control circuitry in the dSPACE board from the higher power converter circuit.

For grid connection, the output of the inverter is filtered with an inductor filter to smooth out the signal and reduce higher order harmonics. The filtered output is then connected to the grid

through a three phase autotransformer. Note that for grid connection, the output voltage of the inverter must be slightly higher than the grid voltage to force current to feed into the grid from the inverter. Also, the frequency of the generated voltages from the inverter must be the same as the frequency of the grid. Therefore, a phase locked loop is used to measure the angle of the grid to synchronize the dq axis rotating reference frame and consequently, the generated AC signals. By synchronizing the dq reference frame with the grid angle, the frequency of the three phase output of the inverter will be equal to the grid frequency with the same angle. While attempting to achieve grid connection, a myriad of problems arose. Some of these problems had solutions while others did not. The three phase grid voltages are assumed to be balanced and constant for the inverter control algorithm design. However, in the case of Lakehead University's facilities, the grid voltage is unbalanced and each phase voltage would vary slightly. As such some DC offset current is present in the AC current fed into the grid. Therefore, more complicated control algorithms or filtering must be used to remove the DC offset before feeding the grid. The main problem with the attempted grid connection is that an expensive three phase isolation transformer is required to isolate the grid from the energy conversion system. The university does not have such equipment and was not able to be purchased due to lack of funding. As a result, grid connection was abandoned and stand-alone DC mode of operation was attempted.

The duty cycle of the buck boost converter is limited to prevent the converter output voltage exceeding the maximum voltage rating of the output capacitor. The voltage rating of the converter's output capacitor is 100V. With an input of 39V (at the maximum power point), the duty cycle should not exceed 0.72. To allow for a safety factor, the maximum allowed duty cycle is set at 0.7. Further, the designed ANFIS is trained offline and the calculations for the duty cycle are completed quickly in real-time. Therefore, the duty cycle calculation performed by the ANFIS does not slow down the system and is of low computational burden for the processor.

Fig. 7-1 shows the GUI for the solar emulator. The irradiance level, and number of PV modules in series and parallel can be changed. Changing the irradiance level will change the maximum power point. Changing the number of PV modules in series will change the voltage rating of the PV panel and changing the number of PV modules in parallel will change the current rating of the PV panel. Thus, the desired maximum power output at an irradiance of $1000W/m^2$ can be easily chosen. For the purposes of this thesis, five PV modules are used in series and 22 modules are used in parallel. This configuration gives a MPP voltage of 39V and

MPP current of 2.05A at 1000W/m^2. Thus, the MPP PV power output at 1000W/m^2 is 80W. Note that if the irradiance, for example, is changed to 300W/m^2 then the maximum power point will be located at 39V and 24W (0.3*80W). The LabVolt power supply acting as the solar emulator is connected to a laptop through USB communication so that the irradiance level can be easily varied. Further, the PV output voltage, current and power can be viewed in the GUI in order to verify operation at the maximum power point for any irradiance level. In real time, the irradiance is varied in the range of 300W/m^2 to 1000W/m^2. Table 7-1 outlines various specifications for the real time system. Fig. 7-2 shows the physical system setup in the lab. Note that the image shows additional equipment that was not used in the experiments for this thesis. Also of note, the solar emulator is not shown in Fig. 7-2 due to its distance from the rest of the circuits. Long banana cables are used to connect the buck boost converter to the solar emulator.

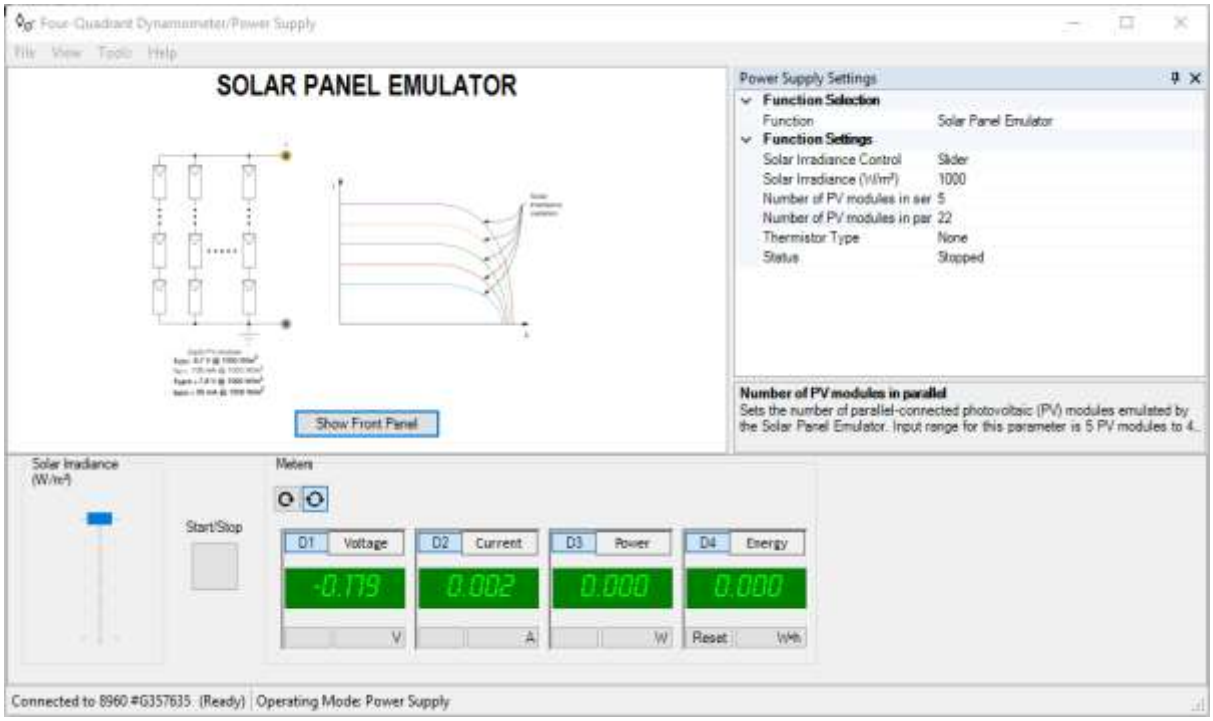

Fig. 7-1: Solar Emulator GUI.

Table 7-1: System specifications

Variable	Value
Solar Emulator	
Number of modules in series	5
Number of modules in parallel	22
I_{mpp} (@1000W/m^2)	2.05A
V_{mpp} (@1000W/m^2)	39V
P_{mpp} (@1000W/m^2)	80W
Buck Boost Converter	
C_{in}	1mF
C_{out}	470µF
MOSFET	IRF520N
Diode	MUR1520
L	2.5mH
R_{load}	100Ω
Current sensor	ACS712

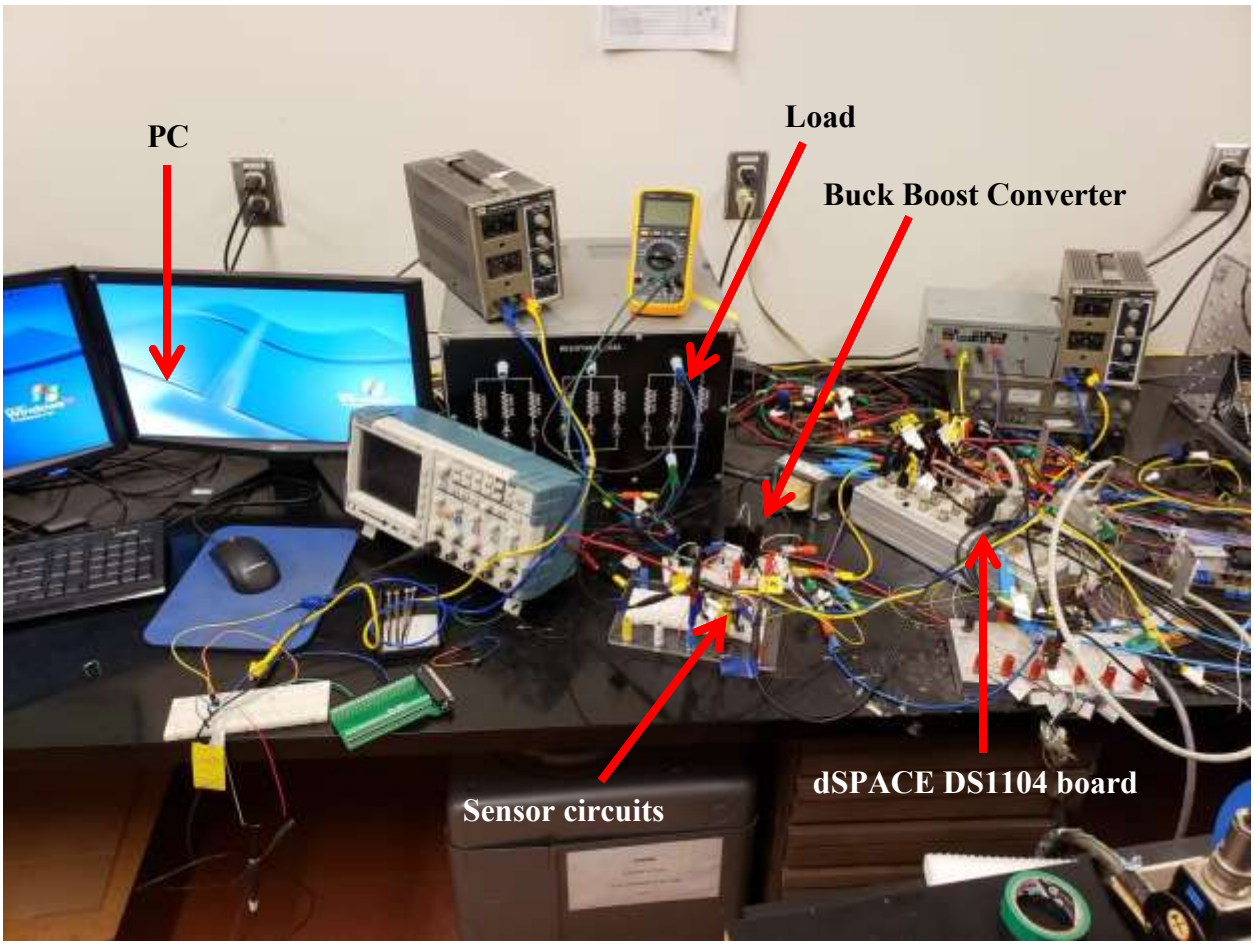

Fig. 7-2: System setup in the lab.

To switch the MOSFET, an external driver circuit with opto-isolator is required. The opto-isolator is used to isolate the low voltage control circuit (dSPACE board) from the high power buck boost converter circuit to prevent damage to the DS1104 board. Fig. 7-3 shows the external driver circuit created using the FOD3180.

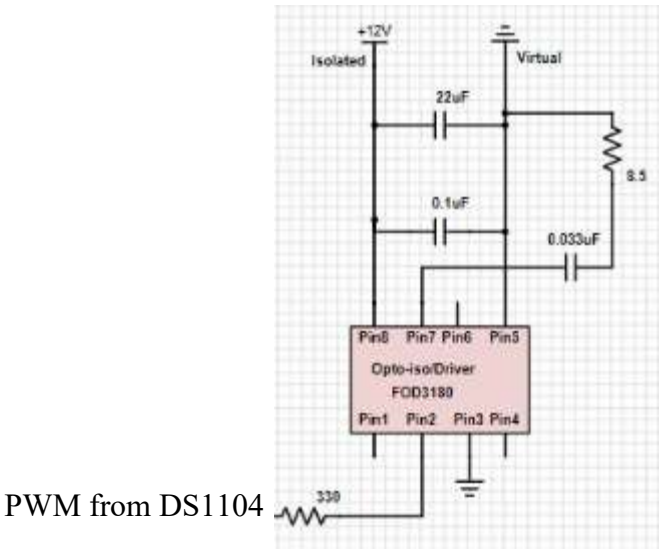

Fig. 7-3: MOSFET driver circuit.

The designed ANFIS is trained using the proposed RLSE-PSO algorithm with real-time system data. The resulting membership functions after training are shown in Fig. 7-4.

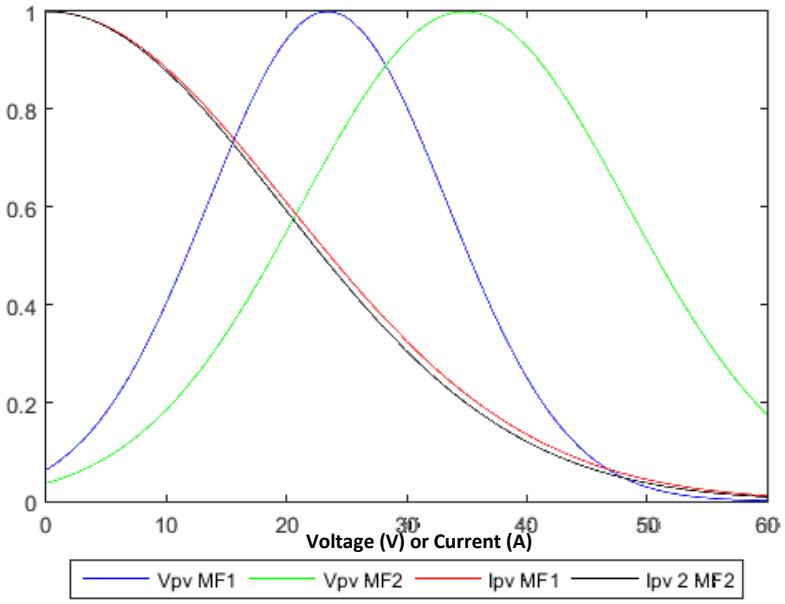

Fig. 7-4: Optimized input membership functions for the real-time system's ANFIS.

The perturb and observe algorithm is first used for MPP extraction to compare the results with the proposed ANFIS-based MPPT controller. The P&O system response at an irradiance of 1000W/m^2 is shown in Fig. 7-5. As such, the maximum power point is 39V, 80W. It is shown

that the P&O algorithm is not effective at tracking the maximum power point. Further, large oscillations are present in the PV output power and current. These oscillations result in major power loss and the inability to effectively track the maximum power point of the solar panel. Note that the system starts running at 5 seconds. Until 5 seconds, the solar panel is connected to an open circuit since the buck boost converter is not switching with PWM pulses. As such, the open circuit voltage of 48V and no output power are measured.

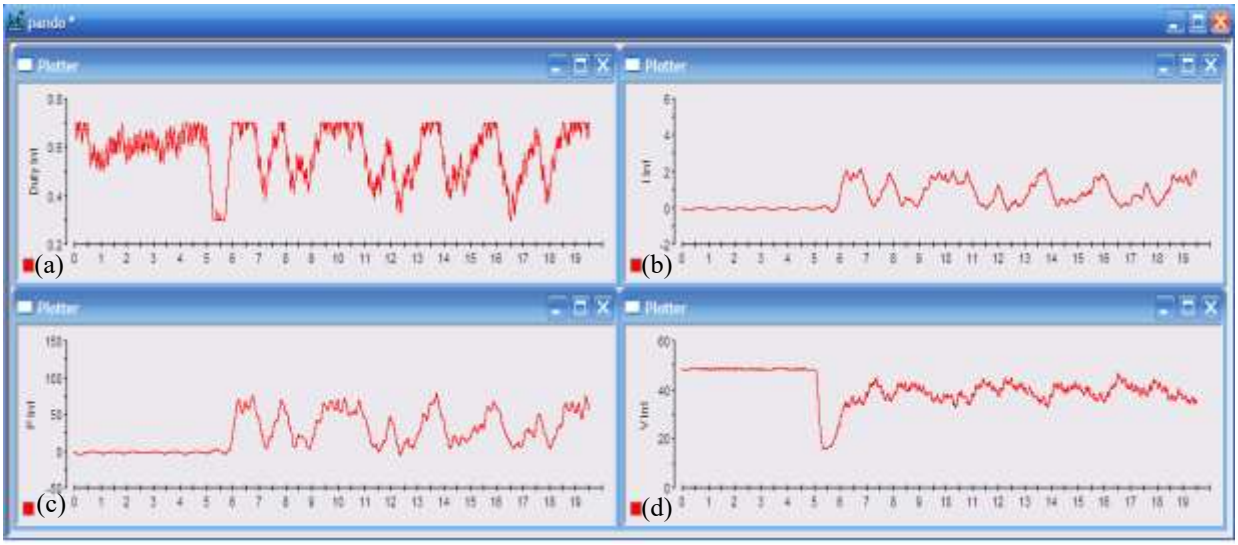

Fig. 7-5: System performance of the P&O-based MPPT controller at 1000W/m^2 (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

Fig. 7-6 shows the system running with the proposed ANFIS-based MPPT controller at an irradiance level of 1000W/m^2. It is shown that the proposed controller effectively forces the solar panel to operate at its maximum power point with no oscillations. Therefore, no power is lost due to oscillations as in the P&O-based system's case. It is proven that the proposed training algorithm and designed ANFIS provides a significantly better response than P&O MPPT algorithm. The duty cycle, PV output voltage, PV output power and PV output current are constant around 69%, 39V, 2.05A and 80W respectively.

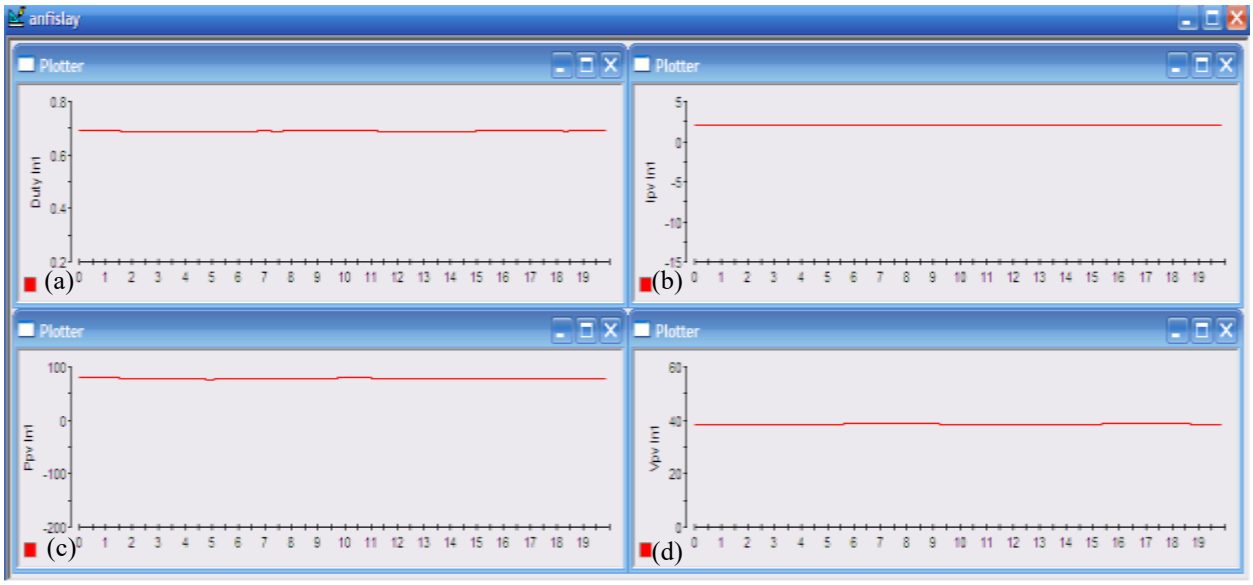

Fig. 7-6: System performance of the ANFIS-based MPPT controller at 1000W/m^2 (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

It is of interest to prove the system can handle different irradiance levels while still maintaining maximum power extraction for a given irradiance condition. Therefore, step changes in irradiance are performed to obtain the system response when the irradiance changes abruptly. Fig. 7-7 shows the ANFIS-based MPPT controller system response with a step-up change from 500W/m^2 to 1000W/m^2. Fig. 7-8 shows the ANFIS-based MPPT controller system response with a step-down change from 1000W/m^2 to 500W/m^2. Fig. 7-9 shows the ANFIS-based MPPT controller system response with a step change from 300W/m^2 to 700W/m^2. From Fig. 7-7, Fig. 7-8 and Fig. 7-9, it is illustrated that the proposed controller effectively forces the solar panel to operate at its MPP regardless of the irradiance level. Further, the system achieves a fast response with no oscillations at the maximum power point regardless of the type of change in irradiance. In Fig. 7-7, the irradiance step change is implemented at 4 seconds. Note that down sampling is used so the time axis (x-axis) for all figures is not accurately representative of one real second. The results provide the illusion that the system response is slow; however, this type of data capture was required since the system response is too fast to view any changes on ControlDesk on the PC. Down sampling must be used to view various measurements and graphs on the PC.

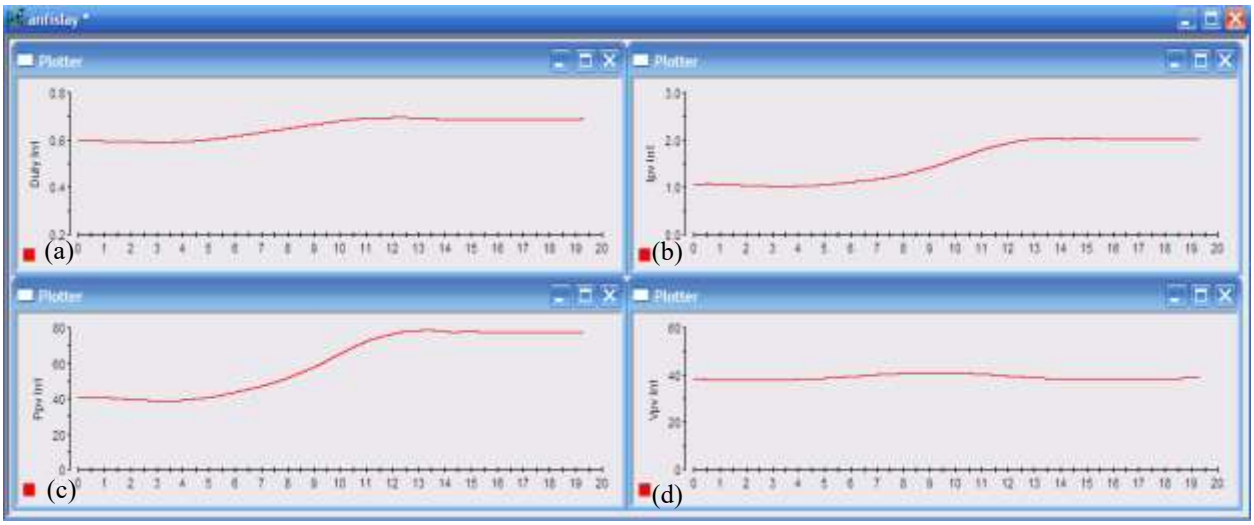

Fig. 7-7: System performance of the ANFIS-based MPPT controller with irradiance step change from 500W/m^2 to 1000W/m^2 at 3 seconds (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

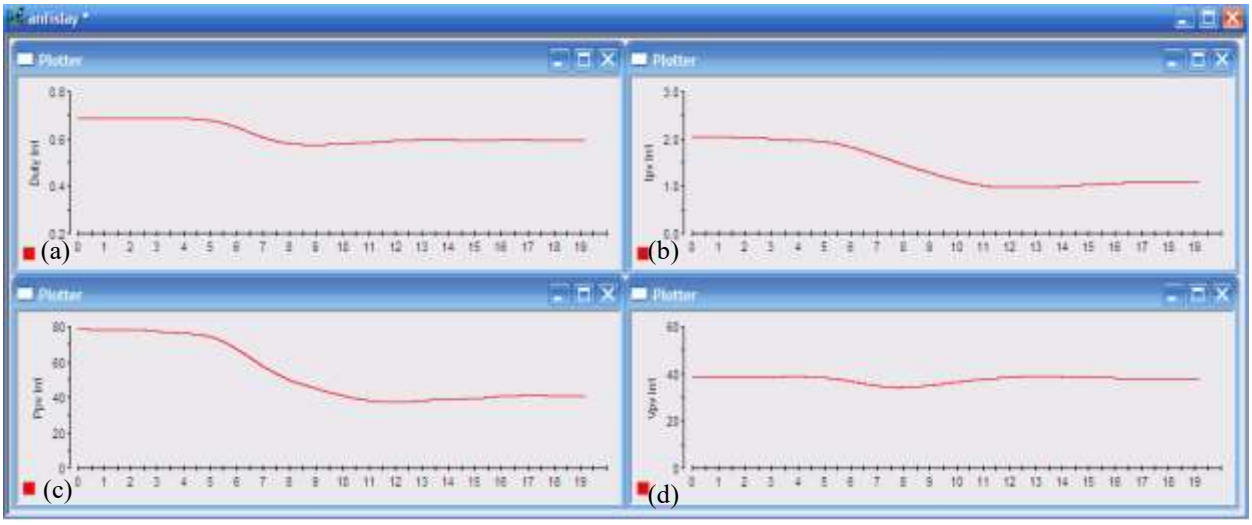

Fig. 7-8: System performance of the ANFIS-based MPPT controller with irradiance step change from 1000W/m^2 to 500W/m^2 at 4 seconds (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

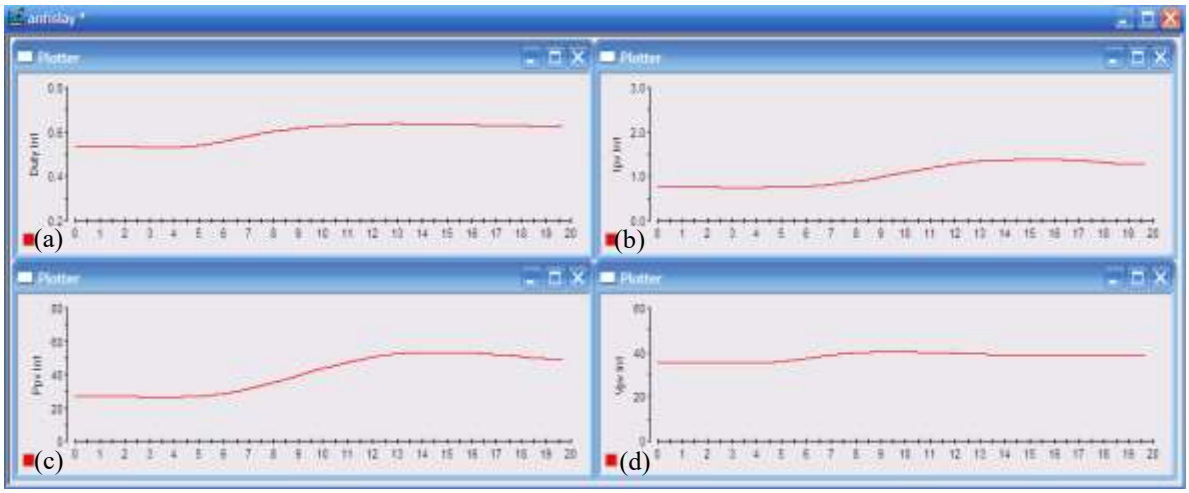

Fig. 7-9: System performance of the ANFIS-based MPPT controller with irradiance step change from 300W/m^2 to 700W/m^2 at 4 seconds (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

Note that all results up to this point were obtained with a 100Ω load on the buck boost converter. Further experiments are performed to verify the proposed controller is able to maintain solar panel MPP operation with a step change in the load. Here, the system is running with a 100Ω load at 1000W/m^2 when the load resistance is abruptly changed to 80Ω. It is shown in Fig. 7-10 that the step change in load resistance does not affect the MPP operation and the proposed controller effectively maintains maximum power extraction from the solar panel.

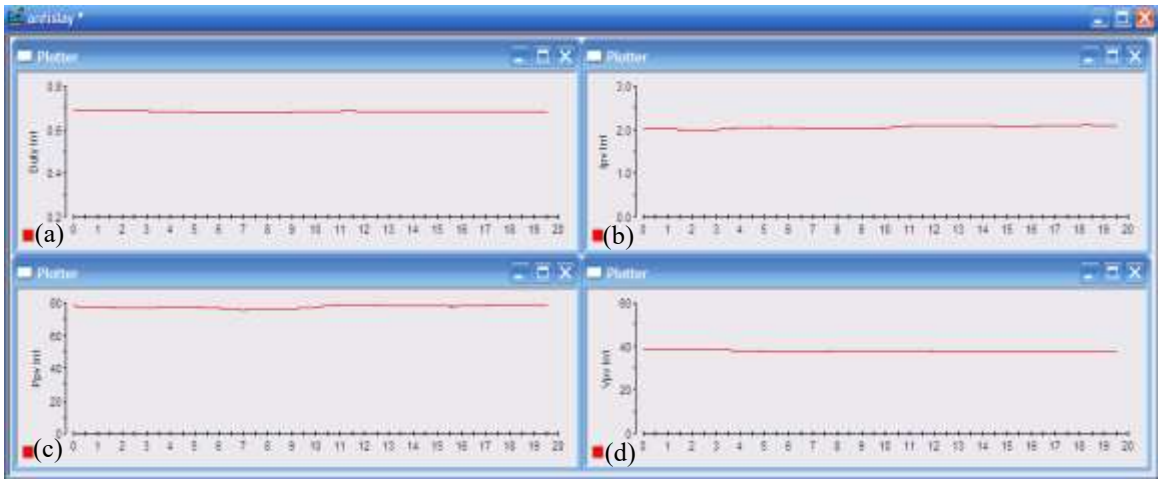

Fig. 7-10: System performance of the ANFIS-based MPPT controller with load step change from 100Ω to 80Ω at 6 seconds (a) duty cycle, (b) solar panel output current, (c) solar panel output power, (d) solar panel output voltage.

Table 7-2 shows a comparison of the theoretical and actual solar power output for the ANFIS-based real-time system trained with the proposed RLSE-PSO algorithm. It is shown that the proposed system is able to very closely extract the maximum possible power without large oscillations from the solar panel regardless of the irradiance level. Therefore, the proposed ANFIS and training algorithm results in higher efficiencies than existing perturb and observe MPPT methods used in industry. The proposed algorithm results in cost savings for the customer as more oscillation-free power is extracted from the PV array. It is further noted that the converter is more efficient at higher irradiance levels since the current is larger. The buck boost converter was designed for an output current higher than the available current when the irradiance level is low. Since the converter was designed for higher currents, it is intuitive that the converter will have a harder time operating the solar panel at its MPP for lower irradiance levels. This fact is verified by viewing the results of the PV power output in table 7-2.

Table 7-2: Comparison of theoretical and actual PV output power for the proposed system

Irradiance (W/m^2)	Theoretical Power (W)	Actual Power (W)
300	24	21
400	32	29
500	40	38
600	48	46
700	56	54
800	64	62
900	72	71.5
1000	80	79

Chapter 8

Summary

An improved ANFIS-based MPPT controller for maximum power extraction from a solar photovoltaic array has been propose. Particularly, the following achievements are provided in this thesis.

- A specific ANFIS-based neuro-fuzzy control scheme was developed to extract maximum power from solar panel to the dc link. The proposed controller was applied to a buck-boost converter as part of the efficient solar energy conversion system.
- A novel RLSE-PSO training algorithm was developed to train the ANFIS system parameters to achieve optimal system performance.

The proposed controller is able to effectively extract maximum power from the PV array for a wide range of irradiance levels and changes in load conditions. The MPPT controller uses the measured PV output voltage and current to calculate a duty cycle for the buck-boost converter corresponding to the maximum power point for a given irradiance and temperature. The temperature of the array is held constant while the irradiance is in the range of 300 W/m^2 to 1000 W/m^2. MATLAB/Simulink was used to simulate the system and train the MPPT controller with the proposed training algorithm. The training algorithm used data obtained from operating the system using the perturb and observe MPPT algorithm. Grid connection was also simulated in which the output of the buck-boost converter was fed into a two level voltage source inverter to create a three phase output for connection with the grid. Excellent results were obtained from the simulation as the proposed controller was able to effectively extract maximum power from the PV array with almost no oscillations around the MPP regardless of irradiance level. Additionally, the inverter control and filter was able to generate a balanced three phase output synchronized with the grid angle and achieved a total harmonic distortion of less than one percent for both of the three phase voltage and current.

To further prove the proposed MPPT controller and proposed RLSE-PSO training algorithm, the system was implemented in real-time using the dSPACE DS1104 DSP board. Grid

connection was attempted; however, lack of proper equipment and funding disallowed for success as a three phase isolation transformer was required. As such, the grid connection was not possible in real-time. For real-time implementation, the buck-boost converter was built and connected to a resistive load. The ANFIS-based MPPT controller was trained using the proposed RLSE-PSO algorithm. Training data was gathered by implementing the P&O MPPT algorithm and saving the measured PV output voltage, current and calculated duty cycle. The effectiveness of the proposed system was verified in real-time as the ANFIS MPPT controller was able to extract maximum power from the PV array for all irradiance levels in the desired range of 300 W/m^2 to 1000 W/m^2 with no oscillations at the MPP. The controller was also able to maintain PV operation at the MPP for a step change in the load resistance. Compared to the P&O MPPT controller, the proposed controller demonstrated superior performance as it was able to extract maximum power from the PV array with little to no power loss due to oscillations. The P&O MPPT algorithm was not able to extract maximum power and created a highly oscillatory response. This thesis provides results that confirm all of the objectives have been completed and system performance extensively substantiated though simulation and real-time experimentation.

8.1 Future Work and Improvements

Through extensive simulations and real time experiments as well as additional theoretical analysis, various areas of improvement have been noted. Possible areas for further investigation are outlined in this section.

The following improvements or changes could be made in future work:

1. Design a snubber circuit for the MOSFET in the buck boost converter to further reduce switching losses.
2. Replace the diode in the converter with another MOSFET. In this scenario, the switching losses that are introduced due to the diode will be significantly less as the ‘on’ resistance of a MOSFET is less than that of a diode. In this situation, the two MOSFETs in the converter need to be synchronized so that when one is on, the other is off to guarantee proper operation of the converter.

3. Experiment with the ANFIS structure by changing the number of rules, the number of input membership functions or the type of membership functions. Varying the ANFIS structure and parameters will change the performance of the MPPT controller. It is possible that better or worse results will be obtained with a differently designed ANFIS based MPPT controller.
4. Replace the PI controllers used for the inverter control in simulation with fuzzy logic controllers or neuro-fuzzy controllers. The more advanced controllers may achieve a better response as they avoid the typical drawbacks of PI controllers.
5. Since the three phase grid voltage is generally unbalanced, include DC current offset calculations in the simulation control algorithm for the inverter to remove DC offset current due to the unbalanced grid.
6. Include DC link voltage control to maintain a relatively constant DC link voltage while still achieving maximum power point tracking of the solar array.
7. Remove the buck boost converter and perform the maximum power point tracking directly with an inverter. Very little research has been completed in this area and would be useful topic to consider for another Master's thesis.
8. Perform further analysis and real time experimentation by applying the proposed MPPT controller to a wind energy conversion system. The proposed algorithm has been applied to a wind energy conversion system in simulation but experimental work was not completed.
9. Complete grid connection or single phase stand-alone operation in real-time with proper equipment.

References

[1] J.-S. R. Jang, C.-T. Sun, and E. Mizutani, *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*. Upper Saddle River, NJ: Prentice Hall, 1997.

[2] F. Karray and C. deSilver, *Soft computing and intelligent systems design: theory, tools, and applications*. Pearson Publishing Inc., 2004.

[3] F. Belhachat, C. Larbes, "Global maximum power point tracking based on ANFIS approach for PV array configurations under partial shading conditions", *Renewable and Sustainable Energy Reviews*, vol. 77, pp. 875-889, 2017.

[4] V.G. Gudise, G.K. Venayagamoorthy, "Comparison of particle swarm optimization and backpropagation as training algorithms for neural networks", *IEEE Swarm Intelligence Symposium*, April 26, pp. 110-117, 2003.

[5] V.Seydi Ghomsheh, M. Aliyari Shoorehdeli , M. Teshnehlab, "Training ANFIS structure with modified PSO algorithm", *Mediterranean Conference on Control and Automation*, July 27-29, 2007.

[6] K. Ishaque, Z. Salam, M.Amjad, S. Mekhilef, "An improved particle swarm optimization (PSO)–based MPPT for PV with reduced steady-state oscillation", *IEEE Trans. on Power Electronics*, vol. 27, no. 8, pp. 3627-3638, August 2012.

[7] T. Esram, P. L. Chapman, "Comparison of photovoltaic array maximum power point tracking techniques", *IEEE Trans. on Energy Conversion*, vol. 22, no. 2, June 2007.

[8] A. El Khateb, N. Abd Rahim, J. Selvaraj and M. N. Uddin, "Fuzzy-logic-controller-based SEPIC converter for maximum power point tracking," *IEEE Trans. on Industrial Applications*, vol. 50, no. 4, pp. 2349-2358, Jul./Aug. 2014.

[9] A.K. Abdelsalam, A.M. Massoud, S. Ahmed, and P.N. Enjeti, "High performance adaptive perturb and observe MPPT technique for photovoltaic-based microgrids," *IEEE Trans. on Power Electronics*, vol. 26, no. 4, pp. 1010-1021, April 2011.

[10] M.A. Gomes de Brito et al. "Evaluation of the main MPPT techniques for photovoltaic applications", *IEEE Trans. on Industrial Electronics*, vol. 60, no. 3, pp. 1156-1167, March 2013.

[11] A. D. Martin, J. R. Vazquez, "MPPT algorithms comparison in PV systems", *IEEE International Conference on Industrial Technology*, pp. 2841-2847, March 2015.

[12] H.A. Sher et al. "A new sensorless hybrid MPPT algorithm based on fractional short circuit current measurement and P&O MPPT", *IEEE Trans. on Sustainable Energy*, vol. 6, no. 4, pp. 1426-1434, October 2015.

[13] B. Subudhi and R. Pradhan, "A comparative study on maximum power point tracking techniques for photovoltaic power systems", *IEEE Trans. on Sustainable Energy*, vol. 4, no. 1, pp. 89-98, January 2013.

[14] H. Abu-Rub et al. "Quasi-z-source inverter-based photovoltaic generation system with maximum power tracking control using ANFIS" *IEEE Trans. on Sustainable Energy*, vol. 4, no. 1, pp. 11-20, January 2013.

[15] M.V. Oliveira, R. Schirru, "Applying particle swarm optimization algorithm for tuning a neuro-fuzzy inference system for sensor monitoring", *Progress in Nuclear Energy*, pp. 177-183, 2009.

[16] Texas Instruments, "Understanding Buck-Boost power stages in switch mode power supplies", *Application Report*, pp. 1-31, November 2002.

[17] Ned Mohan, Tore Undeland and William Robbins, *Power Electronics- Converters, Applications and Design*, John Wiley & Sons Inc., 2003.

[18] Muhammad Rashid, *Power Electronics Handbook*, Elsevier Inc., 2007.

[19] A. Iqbal, H Abu-Rub and Sk. M. Ahmed, "Adaptive neuro-fuzzy inference system based maximum power point tracking of a solar PV module", *IEEE International Energy Conference*, pp. 51-56, 2010.

[20] A. Chikh and A. Chandra, "An optimal maximum power point tracking algorithm for PV systems with climatic parameters estimation", *IEEE Trans. on Sustainable Energy*, vol. 6, no. 2, pp. 644-652, April 2015.

[21] Y. Zou, M. Elbuluk and Y. Sozer, "Stability analysis of maximum power point tracking (MPPT) method in wind power systems", *IEEE Trans. on Industry Applications*, vol. 49, no. 3, pp. 1129-1136, May/June 2013.

[22] P. Garcia et al., "ANFIS-based control of a grid-connected hybrid system integrating renewable energies, hydrogen and batteries", *IEEE Trans. on Industrial Informatics*, vol. 10, no. 2, pp. 1107-1117, May 2014.

[23] Y. Sun et al., “Artificial neural network for control and grid integration of residential solar photovoltaic systems”, *IEEE Trans. on Sustainable Energy*, vol. 8, no. 4, pp. 1484-1495, October 2017.

[24] M. Villalva, J. Gazoli and E. Filho, “Comprehensive approach to modeling and simulation of photovoltaic arrays”, *IEEE Trans. on Power Electronics*, vol. 24, no. 5, pp. 1198-1208, May 2009.

[25] A. Al Nabulsi and R. Dhaouadi, “Efficiency optimization of a DSP based stand-alone PV system using fuzzy logic and dual-MPPT control”, *IEEE Trans. on Industrial Informatics*, vol. 8, no. 3, pp. 573-584, August 2012.

[26] A. Rubaai et al., “Hardware implementation of an adaptive network-based fuzzy controller for DC-DC converters”, *IEEE Trans. on Industry Applications*, vol. 41, no. 6, pp. 1557-1565, November/December 2005.

[27] M. Singh, A. Chandra, “Application of adaptive network-based fuzzy inference system for sensorless control of PMSG-based wind turbine with nonlinear-load-compensation capabilities”, *IEEE Trans. on Power Electronics*, vol. 26, no. 1, pp. 165-175, January 2011.

[28] C. Chiu, Y. Ouyang, “Robust maximum power tracking control of uncertain photovoltaic systems: a unified T-S fuzzy model-based approach”, *IEEE Trans. on Control Systems Technology*, vol. 19, no. 6, pp. 1516-1526, November 2011.

[29] W. Lin, C. Hong and C. Chen, “Neural-network-based MPPT control of a stand-alone hybrid power generation system”, *IEEE Trans. on Power Electronics*, vol. 26, no. 12, pp. 3571-3581, December 2011.

[30] N. Femia, G. Petrone, G. Spagnulolo and M. Vitelli, “Optimization of Perturb and Observe maximum power point tracking algorithm”, *IEEE Trans. on Power Electronics*, vol. 20, no. 4, pp. 963-973, July 2005.

[31] C. Chiu. “T-S fuzzy maximum power point tracking control of solar power generation systems”, *IEEE Trans. on Energy Conversion*, vol. 25, no. 4, pp. 1123-1132, December 2010.

[32] P. Krause, O. Wasynczuk, S. Sudhoff and S. Pekarek, *Analysis of Electric Machinery and Drive Systems*. John Wiley & Sons, Inc. 2013.

[33] R. Koad, A. F. Zobaa et al., “A Novel MPPT Algorithm Based on Particle Swarm Optimization for Photovoltaic Systems”, IEEE Trans. Sustainable Energy, vol. 8, no.2, pp. 468-476, April 2017.

[34] S. Mohanty, B. Subudhi, P. Ray, “A New MPPT Design Using Grey Wolf Optimization Technique for Photovoltaic System Under Partial Shading Conditions”, IEEE Trans. Sustainable Energy, vol. 7, no. 1, pp. 181-188, Jan. 2016.

[35] K. Amara et. al., “Improved Performance of a PV Solar Panel with Adaptive Neuro Fuzzy Inference System ANFIS based MPPT”, 2018 7th International Conference on Renewable Energy Research and Applications (ICRERA), Oct. 2018.

[36] El Hadji Mbaye Ndiaye et al. “Adaptive Neuro-Fuzzy Inference System Application for The Identification of a Photovoltaic System and The Forecasting of Its Maximum Power Point”, 2018 7th International Conference on Renewable Energy Research and Applications (ICRERA), Oct. 2018.

[37] S. Benhalima et al. “Real-time experimental implementation of an LMS-adaline-based ANFIS controller to drive PV interfacing power system”, IET Renewable Power Generation, vol. 13, issue 7, pp. 1142-1152, May 2019.

[38] A. Rezaee Jordehi “Parameter selection in particle swarm optimisation: a survey”, Journal of Experimental & Theretical Artificial Intelligence, pp. 527-542, June 2013.

Appendix A

Source Code

The MATLAB code for the perturb and observe algorithm as well as the code for the proposed ANFIS training method is provided. The code for the ANFIS is also included.

The code for the P&O algorithm is shown below:

```
1.  function D = PandO(Vpv, Ipv)
2.
3.  persistent Dprev Pprev Vprev %save values of each function call
4.
5.  %first run
6.  if isempty(Vprev)
7.      Vprev=0;
8.      Pprev=0;
9.      Dprev=0.2; %start at duty of 0.5
10. end
11.
12. %change in duty cycle and duty limits
13. deltaD=0.0001;
14. minD=0.1;
15. maxD=0.9;
16.
17. %power and delta calculations
18. Ppv=Vpv*Ipv;
19. deltaV=Vpv-Vprev;
20. deltaP=Ppv-Pprev;
21.
22. %P&O algorithm
23. if deltaP~=0
24.     if deltaP>0
```

```
        if deltaV<0
            D=Dprev+deltaD;
        else
            D=Dprev-deltaD;
        end
    else
        if deltaV<0
            D=Dprev-deltaD;
        else
            D=Dprev+deltaD;
        end
    end
else
    D=Dprev;
end

%limit duty cycle
if D<minD
    D=Dprev;
end
if D>maxD
    D=Dprev;
end

%save values
Vprev=Vpv;
Pprev=Ppv;
Dprev=D;
end
```

The code for the proposed RLSE-PSO training method is shown below:

```
tic %time the trainimg

train_num=2358; % training data pairs
```

```
Emax=0.15; %allowed root mean square epoch error
epochs=120; %max number of training epochs
epoch_num=1; %start at first epoch
E=0; %initialize error
RMSE=0;%initialize root mean square error
numMF=4;%num of membership functions
numPop=16;%number of members in population
w=1;
lambda_w=0.98;%damping factor for w
c1=2.3;%personal learning rate 2
c2=2.9;%global learning rate 2.9
globalbestError=inf;%initial error for the best member
e=zeros(1,train_num);%initialize error vector

%max and min values for center of Vpv MFs
maxposV=250;
minposV=0;
%max and min values for center of I MFs
maxposI=13;
minposI=0;
%max and min values for std dev
maxdev=90;
mindev=0;

for j=1:numMF:numPop*numMF
    %initial population position (4MFs, center, sigma)
    %mfpos=[c,sigma]
    mfpos(j,1)=minposV+(maxposV-minposV)*rand; %initialize V center
    mfpos(j+(numMF-3),1)=minposV+(maxposV-minposV)*rand;

    mfpos(j+2,1)=minposI+(maxposI-minposI)*rand; %initialize I center
    mfpos(j+(numMF-1),1)=minposI+(maxposI-minposI)*rand;

end
mfpos(:,2)=unifrnd(mindev,maxdev,numMF*numPop,1); %initialize std dev

```

```
41. globalbestV=unifrnd(minposV,maxposV,numMF-numMF/2,1);%initialize      to
random values
42. globalbestI=unifrnd(minposI,maxposI,numMF-numMF/2,1);%initialize      to
random values
43. globalbestdev=unifrnd(mindev,maxdev,numMF,1);
44. globalbest(1:2,1)=globalbestV;
45. globalbest(3:4,1)=globalbestI;
46. globalbest(:,2)=globalbestdev;
47. vel=zeros(numMF*numPop,2); %initialize velocity
48. mf_bestpos=zeros(numMF*numPop,2);
49.
50. %initialize linear consequent parameters to some arbitrary value
51. theta=zeros(12,1); %theta=[p1,q1,r1,p2,q2,r2,p3,q3,r3,p4,q4,r4]'
52. P=1000*eye(12); %initial P matrix for recursive LSE
53.
54. while epoch_num<=epochs
55.     E=0; %reset error after each epoch
56.     %use globalbest for LSE MF parameters
57.     for i=1:train_num
58.
59.         %extract training data
60.         Vpv=train_data(i,1);
61.         Ipv=train_data(i,2);
62.         des_duty=train_data(i,3);
63.
64.         %Layer 1 outputs (MF outputs with singleton input)
65.         O11=gauss_MF(Vpv,globalbest(1,1),globalbest(1,2));
66.         O12=gauss_MF(Vpv,globalbest(2,1),globalbest(2,2));
67.         O21=gauss_MF(Ipv,globalbest(3,1),globalbest(3,2));
68.         O22=gauss_MF(Ipv,globalbest(4,1),globalbest(4,2));
69.
70.         %Layer 2 outputs (product t-norm to obtain firing strengths)
71.         W1=O11*O21;
72.         W2=O11*O22;
73.         W3=O12*O21;
74.         W4=O12*O22;
75.
```

```
76.         %Layer 3 outputs (normalize firing strengths)
77.         W1_bar=W1/(W1+W2+W3+W4);
78.         W2_bar=W2/(W1+W2+W3+W4);
79.         W3_bar=W3/(W1+W2+W3+W4);
80.         W4_bar=W4/(W1+W2+W3+W4);
81.
82.         %recursive least squares estimator to estimate linear
parameters
83.
ap=[W1_bar*Vpv;W1_bar*Ipv;W1_bar;W2_bar*Vpv;W2_bar*Ipv;W2_bar;W3_bar*Vpv;W3
_bar*Ipv;W3_bar;W4_bar*Vpv;W4_bar*Ipv;W4_bar];
84.         P=P-(P*(ap*ap')*P)/(1+ap'*P*ap);
85.         theta=theta+P*ap*(des_duty-ap'*theta);
86.
87.         %Layer 4 outputs (output of each Rule)
88.         O1=W1_bar*(theta(1)*Vpv+theta(2)*Ipv+theta(3));
89.         O2=W2_bar*(theta(4)*Vpv+theta(5)*Ipv+theta(6));
90.         O3=W3_bar*(theta(7)*Vpv+theta(8)*Ipv+theta(9));
91.         O4=W4_bar*(theta(10)*Vpv+theta(11)*Ipv+theta(12));
92.
93.         %Layer 5 output (output of ANFIS)
94.         xout=O1+O2+O3+O4;
95.
96.         %new error calculation
97.         E=xout-des_duty;
98.         e(i)=E;
99.
100.             %PSO for nonlinear MF parameters
101.             for  j=1:numMF:numPop*numMF  %extract each member from
population
102.                 %error for member's position
103.                 posError=obj_func(Vpv,Ipv,des_duty,mfpos(j:(j+(numMF-
1)),:),theta);
104.
105.                 %if first epoch and first data point initialize
bestpos error
106.                 %else calculate error
```

```
107.                    if i==1 && epoch_num==1
108.                        bestposError=inf;
109.                    else
110.
bestposError=obj_func(Vpv,Ipv,des_duty,mf_bestpos,theta);
111.                    end
112.
113.                    %global best position error (i.e. best member's
error)
114.
globalbestError=obj_func(Vpv,Ipv,des_duty,globalbest,theta);
115.
116.                    %check for best position for each member and best
position
117.                    %among all members
118.                    if posError<bestposError
119.                        mf_bestpos(j:(j+(numMF-1)),:)=mfpos(j:(j+(numMF-
1)),:);
120.                    end
121.                    if bestposError<globalbestError
122.                        globalbest=mf_bestpos(j:(j+(numMF-1)),:);
123.                    end
124.
125.                    %update velocity and position. dampen w
126.                    vel(j:(j+(numMF-1)),:)=w.*vel(j:(j+(numMF-
1)),:)+c1.*rand(4,2).*(mf_bestpos(j:(j+(numMF-1)),:)-mfpos(j:(j+(numMF-
1)),:))+c2.*rand(4,2).*(globalbest-mfpos(j:(j+(numMF-1)),:));
127.                    mfpos(j:(j+(numMF-1)),:)=mfpos(j:(j+(numMF-
1)),:)+vel(j:(j+(numMF-1)),:);
128.                    w=w*lambda_w;
129.
130.                    %limit the position of the members (c,sigma) to a
range
131.                    %suitable for the problem
132.                    mfpos(j:(j+(numMF-3)),1)    =    max(mfpos(j:(j+(numMF-
3)),1),minposV); %lower V limit
```

```
                    mfpos(j:(j+(numMF-3)),1)    =    min(mfpos(j:(j+(numMF-3)),1),maxposV); %upper V limit

                    mfpos(j+2:(j+(numMF-1)),1)  =  max(mfpos(j+2:(j+(numMF-1)),1),minposI); %lower I limit
                    mfpos(j+2:(j+(numMF-1)),1)  =  min(mfpos(j+2:(j+(numMF-1)),1),maxposI); %upper I limit

                    mfpos(j:(j+(numMF-1)),2)    =    max(mfpos(j:(j+(numMF-1)),2),mindev); %lower std dev limit
                    mfpos(j:(j+(numMF-1)),2)    =    min(mfpos(j:(j+(numMF-1)),2),maxdev); %upper std dev limit

                end

            end

            RMSE(epoch_num)=sqrt(mean(e(:).^2)); %save rmse
            epoch_num=epoch_num+1; %increment epoch

            %stop training if error becomes less than threshold
            if RMSE<Emax
               break
            end
        end

        figure
        plot(RMSE,'r') %plot error
        xlabel('Epoch Number');
        ylabel('Error');
        title('Error vs Epochs');

        toc
```

The code for the designed ANFIS is shown below:

```
function D = ANFIS(Vpv, Ipv)

    %nonlinear and linear parameters from training output
globalbest=[11.9042987194588,38.0017037607356;80,50;11,0.899486922983646;11,20.3333988146082];

theta=[0.303482362634626;3.51784508603727;2.01702791374873;2.2183738183
1105;1.13923299637881;-227.911375544320;0.0803414517982133;-
0.406745853735156;-3.97704139448619;-0.138821510109730;-
0.0559088032518080;14.9424816182171];

    %duty cycle limits
         Dmax=0.9;
         Dmin=0.1;

         %Layer 1 outputs (MF outputs with singleton input)
         O11=gauss_MF(Vpv,globalbest(1,1),globalbest(1,2));
         O12=gauss_MF(Vpv,globalbest(2,1),globalbest(2,2));
         O21=gauss_MF(Ipv,globalbest(3,1),globalbest(3,2));
         O22=gauss_MF(Ipv,globalbest(4,1),globalbest(4,2));

         %Layer 2 outputs (product t-norm to obtain firing strengths)
         W1=O11*O21;
         W2=O11*O22;
         W3=O12*O21;
         W4=O12*O22;

         %Layer 3 outputs (normalize firing strengths)
         W1_bar=W1/(W1+W2+W3+W4);
         W2_bar=W2/(W1+W2+W3+W4);
         W3_bar=W3/(W1+W2+W3+W4);
         W4_bar=W4/(W1+W2+W3+W4);

         %Layer 4 outputs (output of each Rule)
```

```
        O1=W1_bar*(theta(1)*Vpv+theta(2)*Ipv+theta(3));
        O2=W2_bar*(theta(4)*Vpv+theta(5)*Ipv+theta(6));
        O3=W3_bar*(theta(7)*Vpv+theta(8)*Ipv+theta(9));
        O4=W4_bar*(theta(10)*Vpv+theta(11)*Ipv+theta(12));

        %Layer 5 output (output of ANFIS)
        D=O1+O2+O3+O4;

        %apply duty cycle limits
        D=max(D,Dmin);
        D=min(D,Dmax);

    end
```

Appendix B

Block Diagrams

Block diagrams from the simulation and real time implementation are presented.

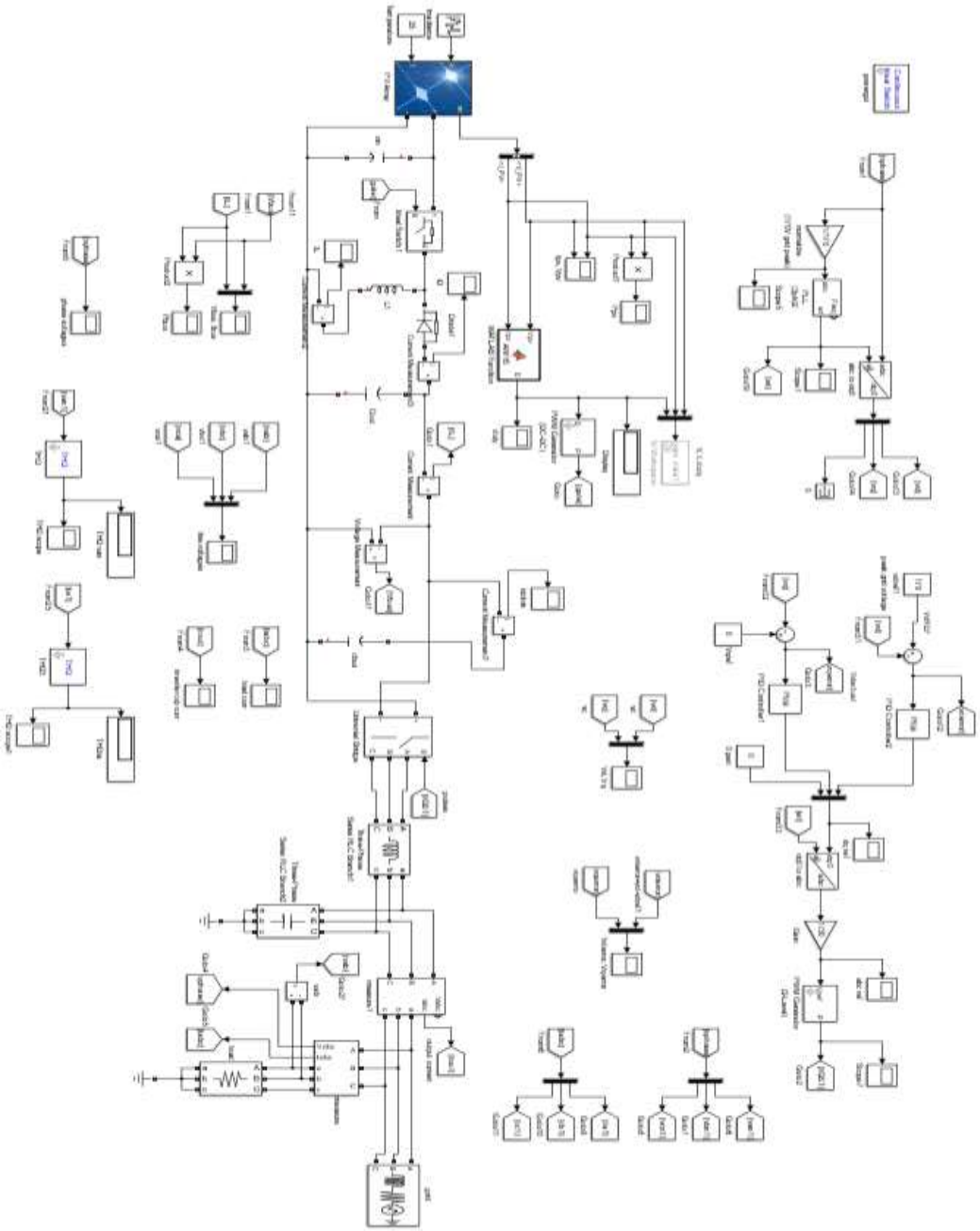

Fig. B-1: Full-scale system simulation block diagram with ANFIS-based MPPT controller.

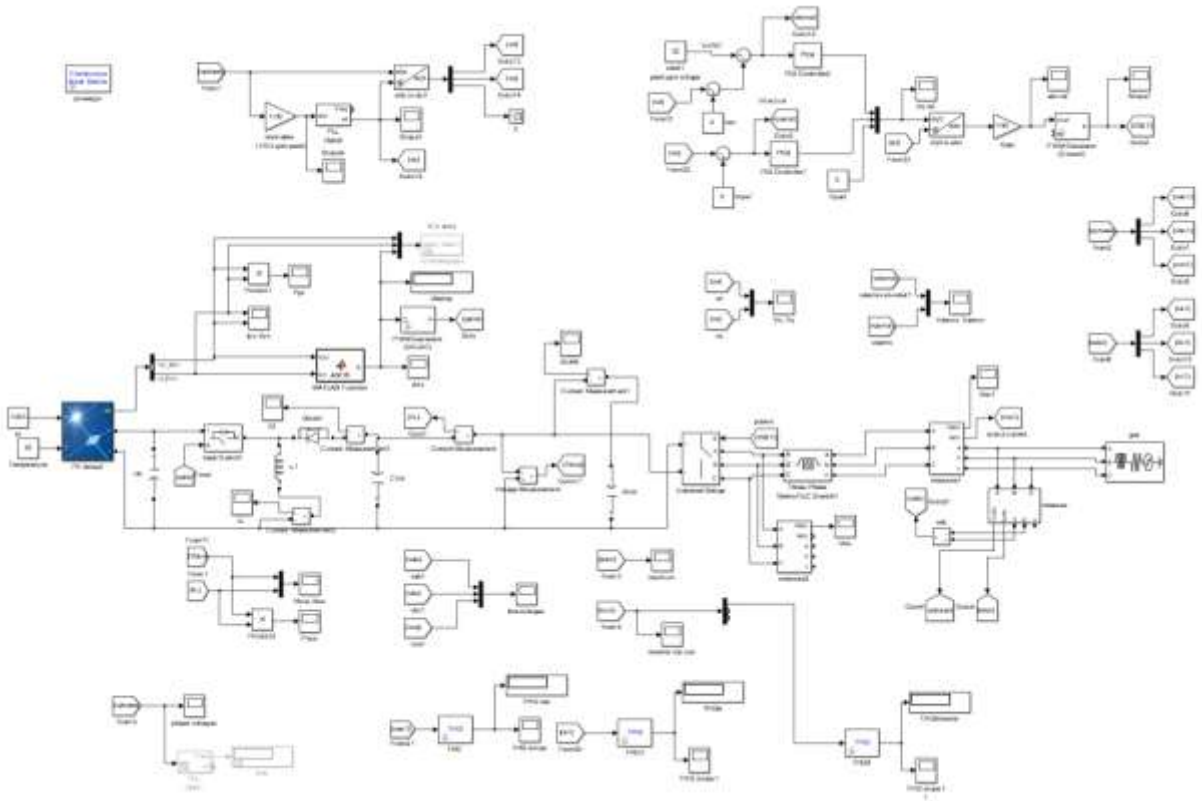

Fig. B-2: Grid connected real time system simulation block diagram with ANFIS-based MPPT controller.

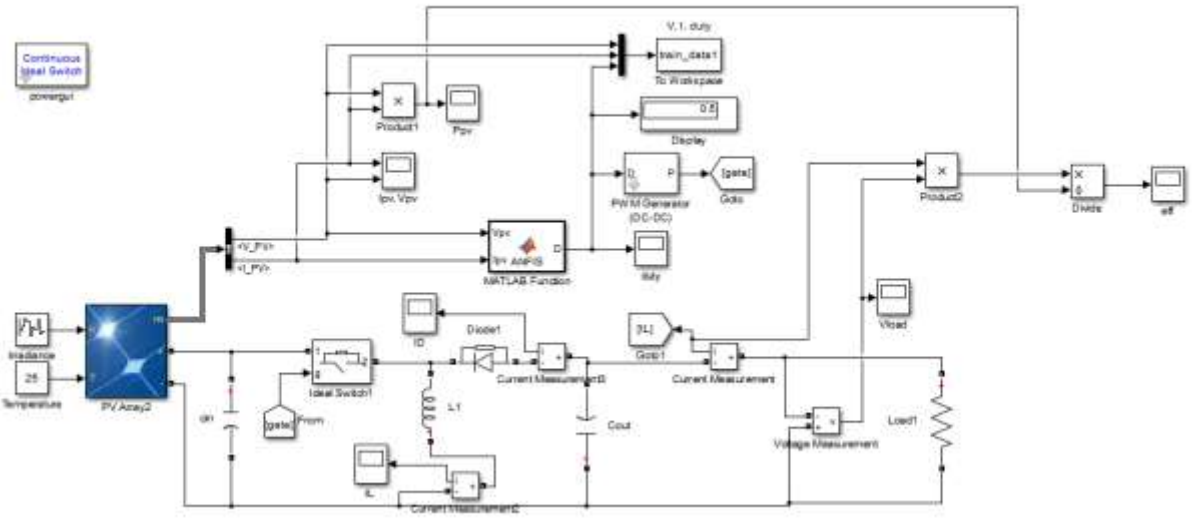

Fig. B-3: Simulink block diagram for the system with 100Ω DC load.

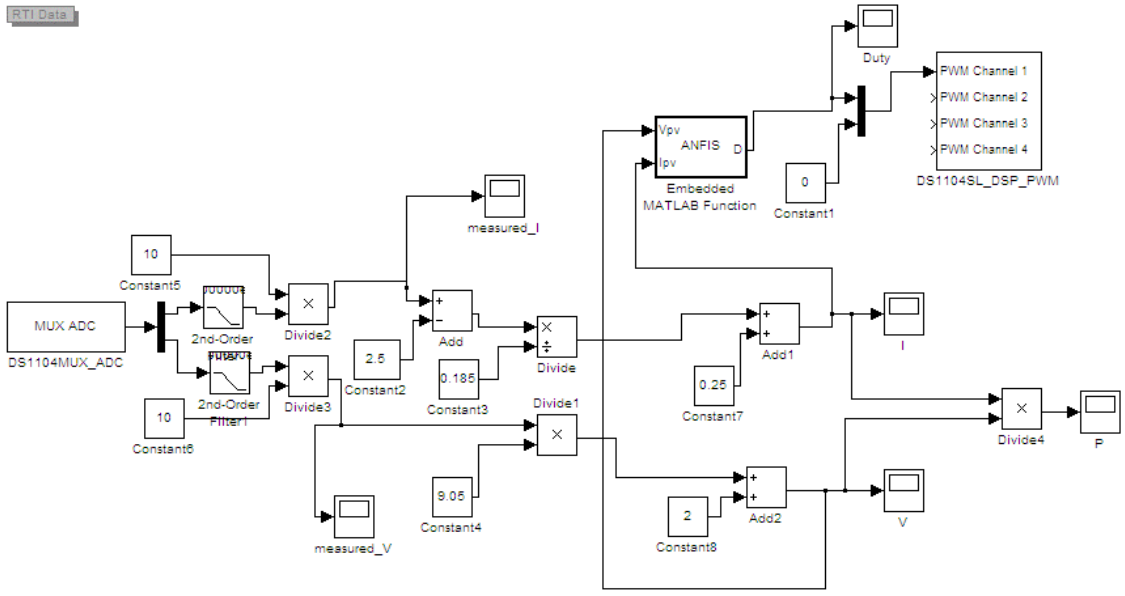

Fig. B-4: Real time implementation block diagram.

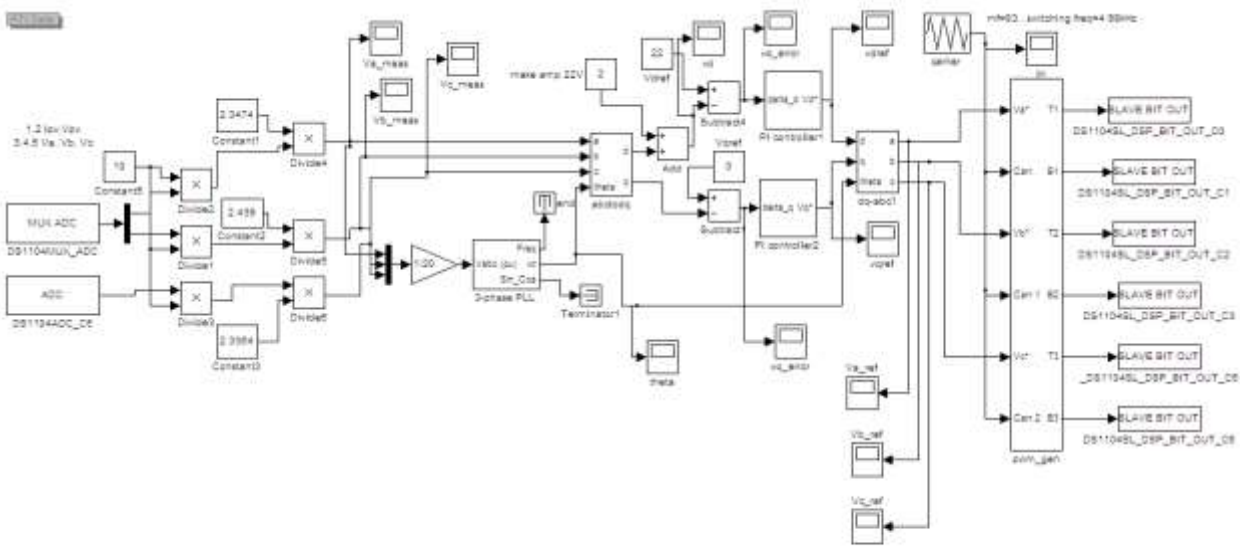

Fig. B-5: Real-time grid connected inverter control block diagram.

Note that the control scheme in Fig. B-5 effectively controls the inverter. Grid connected results, however, cannot be provided due to the lack of a three phase isolation transformer needed to achieve grid connection.

Appendix C

LabVolt Series 8960-20 Solar Emulator Module Configuration

In order to use the solar emulator or the wind turbine emulator as a built in function to the LabVolt dynamometer, there are certain steps that must be taken for proper operation. The list below outlines the steps to follow:

1. Install LVDAC-EMS from LabVolt's website and ensure that the associated drivers install correctly with no malfunctions
2. Turn the dynamometer on and put it in power supply mode for the solar emulator or dynamometer mode for the wind turbine emulator
3. Connect the dynamometer to the computer with the black USB cable provided. Note: Do NOT use a USB extension cable as it will cause improper operation and a myriad of issues.
4. Open LVDAC-EMS application on the computer
5. Choose the local ac voltage and frequency then click OK
6. Click Dynamometer on the menu bar at the top of the window and choose Four-quadrant dynamometer/ Power Supply
7. A new window will open with all the features installed on the module
8. To use the wind turbine or solar emulator, click the function drop down menu and choose the appropriate function
9. The window will update with the various features and information regarding the emulator
10. Change the associated parameters for your application and click Start to run the emulator and obtain an output from the module

Appendix D

DC Link Current Surge Results

It is of interest to consider the effects of a current surge in the dc link. Generally, the surge current is considered to be three times that of current under normal operating conditions. Under normal operating conditions for the full-scale system, the dc link current is around a maximum of 10A. As such, 30A of surge current is introduced in Simulink simulations to verify the operation of the proposed ANFIS-based MPPT controller. A step input of 30A is applied at time 0.166s for the duration of 0.166s. As shown in the figures below, the dc link has a step in current of 30A at 0.166s. Also shown is the solar panel output power. It is proven that even with the large dc link current surge, the proposed MPPT controller is able to extract maximum power from the PV array in the same manner as compared to normal operating conditions. Both of the three phase output currents and voltages at the inverter end have a total harmonic distortion less than 5% as per IEEE standards. The simulation results of a dc link current surge show that the proposed RLSE-PSO training algorithm and designed ANFIS-based MPPT controller can handle a wide range of abnormal operating conditions anywhere in the system.

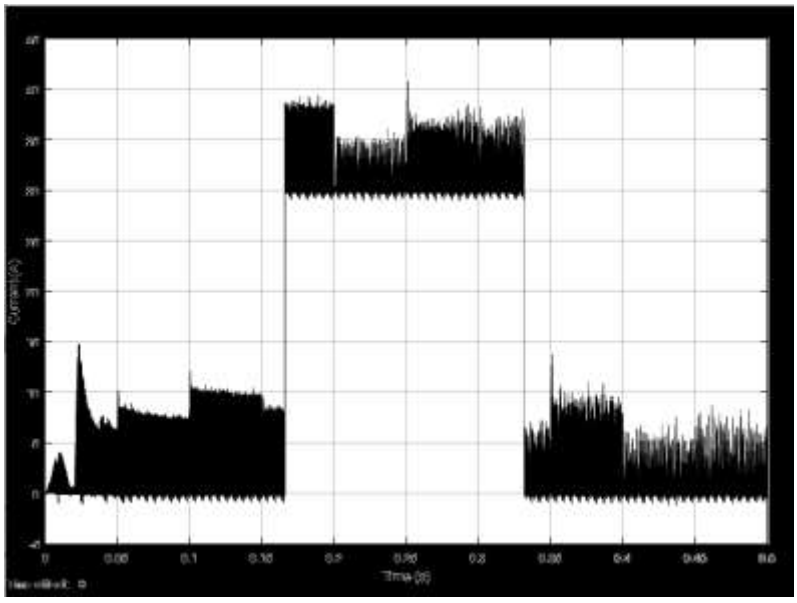

Fig. D-1: DC link current with dc link current surge of 30A.

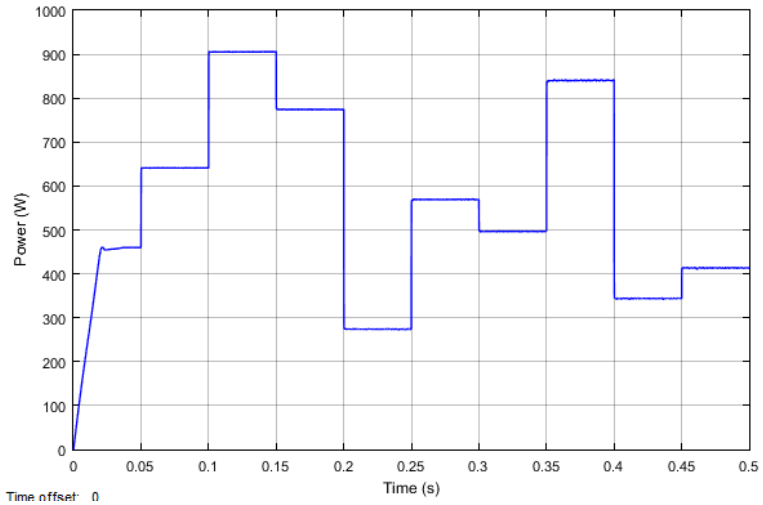

Fig. D-2: PV output power with dc link current surge of 30A.

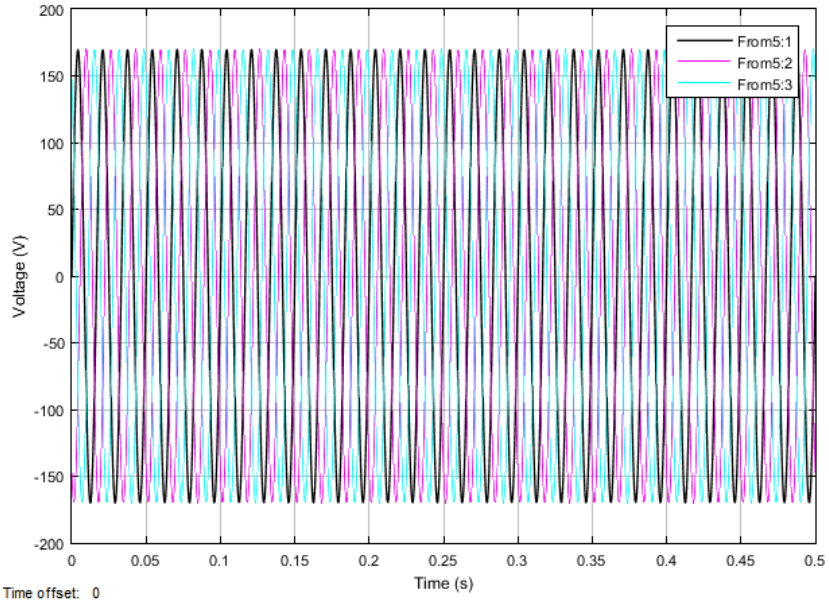

Fig. D-3: Phase voltages at inverter end with dc link current surge of 30A.

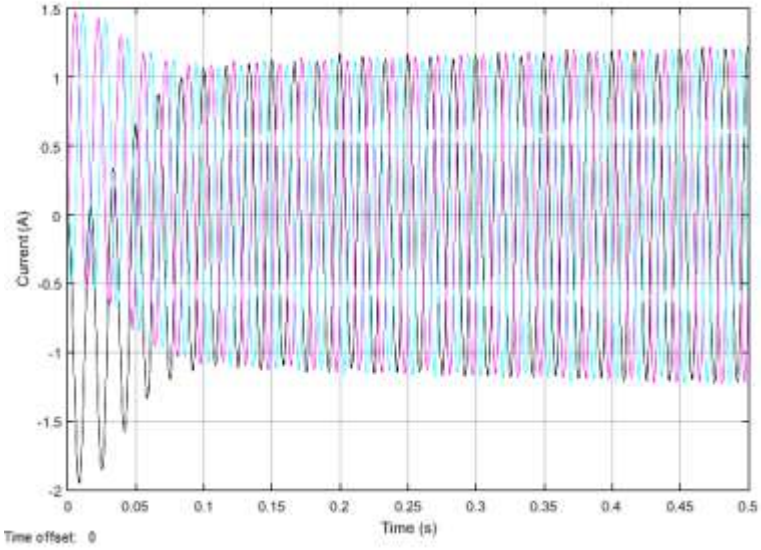

Fig. D-4: Currents at inverter end with dc link current surge of 30A.

Appendix E

Associated Publications

Published:

[1] Jeffrey Andrew-Cotter, M. Nasir Uddin, Ifte Khairul Amin, “Particle Swarm Optimization based Adaptive Neuro-Fuzzy Inference System for MPPT Control of a Three-Phase Grid-Connected Photovoltaic System”, *IEMDC 2019*, 2019.

Submitted:

[2] Ifte Khairul Amin, Mohammad Uddin, Jeffrey Andrew-Cotter, “PSO-ANFIS based MPPT Control of Hybrid Renewable Power Generation for Remote Area Power Supply”, IEEE IAS Annual Meeting, 2019, Mexico City, Mexico.