

LB/TH/33/2025
TH5914

**MACHINE LEARNING BASED MAXIMUM POWER POINT
TRACKING**

Kaluthara Wedage Canish Nauka Shalendra Fonseka

218523G

Master of Science in Artificial Intelligence

Department of Computational Mathematics
Faculty of Information Technology

University of Moratuwa
Sri Lanka

Apr 2024

MACHINE LEARNING-BASED MAXIMUM POWER POINT TRACKING

Kaluthara Wedage Canish Nauka Shalendra Fonseka

218523G

Thesis/Dissertation submitted in partial fulfillment of the requirements for the degree
Master of Science in Artificial Intelligence

Department of Computational Mathematics
Faculty of Information Technology

University of Moratuwa
Sri Lanka

Apr 2024

DECLARATION

I declare that this is my own work and this thesis / dissertation does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other University or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text. I retain the right to use this content in whole or part in future works (such as articles or books).

Signature:

Date: 27th April 2024

The above candidate has carried out research for the PhD/ MPhil/ Masters thesis/ dissertation under my supervision. I confirm that the declaration made above by the student is true and correct.

Name of Supervisor: Dr. (Mrs.) Thushari Silva, PhD

Signature of the Supervisor:

Date: 27/04/2024

DEDICATION

I dedicate this thesis to the most remarkable persons who have played key roles in shaping my academic journey. First and foremost, I would like to convey my heartfelt gratitude to my dear beloved wife and cherished family members whose unwavering support and encouragement have been my constant motivation. Their belief in me during the challenging times has been a beacon of strength.

I extend my deepest appreciation to my esteemed supervisor, Dr. (Mrs.) Thushari Silva, PhD, and the academic faculty for their invaluable guidance, expertise, and mentorship. Their trust in my abilities, constructive feedback, and insightful critiques have significantly contributed to the refinement of my work and my growth as a researcher. I am also indebted to my friends whose friendship and unwavering support have been instrumental in keeping me inspired and focused throughout this journey.

Additionally, I express my sincere gratitude to the Sri Lanka Navy for providing me with the opportunity to pursue this degree while serving my nation. Their support and guidance have been indispensable in the completion of this research.

This thesis stands as a testament to the collective efforts of those mentioned above. I am indeed deeply grateful for their presence in my life journey and for their instrumental role in helping and supporting me to achieve this significant milestone.

ACKNOWLEDGEMENT

I would like to express my sincere gratitude to everyone whose steadfast assistance was necessary to finish this thesis.

First and foremost, my gratitude goes to the supervisor, Dr. (Mrs.) Thushari Silva, PhD, whose unwavering support, guidance, encouragement which have shaped the trajectory of this research. Her expertise and insights have significantly enhanced the quality of this work. I am indebted to the dedicated academic staff at the Faculty of Information Technology, University of Moratuwa, whose commitment to excellence has challenged me to grow as a Master's student. Their passion for their fields has served as a constant wellspring of inspiration.

I express sincere thanks to the Sri Lanka Navy for their invaluable support, understanding, and flexibility, enabling me to pursue this degree while fulfilling my duties to my country.

My heartfelt appreciation also goes to my beloved family members and friends, whose boundless love, care, encouragement and specially belief in me have been unwavering. Your sacrifices and encouragement have been the cornerstone of my journey, and I am profoundly grateful for your steadfast support.

Further, I extend my gratitude to all the participants whose contributions and willingness to share their experiences have made this research possible. I want to express my sincere gratitude and appreciation to each of you. I am honored to have had you by my side during this journey, and this thesis is a monument to your unfailing support and direction.

ABSTRACT

Origin of the solar cell dated back to 1883 when the first solar power generation was invented. A solar system converts sunlight into electrical energy through photovoltaic panels and known as solar power generation. At present solar power generation shows highest development of among all renewable resources. Maximum Power Point Tracking is a method of optimizing the solar panel in order to obtain maximum power output from the solar panel. Comprehensive literature review of MPPT systems, tracing their historical development and examining conventional and modern MPPT methods were carried out and observed existing MPPT methods often fail to capture the relationships between voltage, current, irradiance, and temperature parameters which directly affect the optimal operating point of a photovoltaic panel without disturbing the energy harvesting process, to deliver the maximum power output of the photovoltaic system. As a result, the overall efficiency and performance of the photovoltaic system may be compromised.

The proposed solution involves leveraging the capabilities of machine learning approaches to learn the complex relations between the voltage, current, irradiance and temperature parameters using random forest regression and the proposed approach aims to overcome the limitations of traditional MPPT to obtain optimized output power from photo-voltic system. Developed MPPT model is tested using real-world data collected from PV installations under diverse environmental conditions. The tests evaluate the model's accuracy, adaptability, and performance compared to conventional MPPT techniques. Results demonstrate the effectiveness of the machine learning-based approach in improving energy harvesting efficiency and overall system performance. The findings suggest that machine learning-based VCIT-MPPT offers a promising solution for optimizing PV system performance and maximizing energy yields. In conclusion, this thesis presents a novel approach to MPPT leveraging machine learning techniques named as VCIT-MPPT, showing significant improvement in efficiency of energy harvesting and MPPT system performance. The findings contribute to the advancement of renewable energy technologies and pave the way for more efficient utilization of solar power resources.

Keywords: Maximum Power Point Tracking (MPPT), Photovoltaic (PV), Machine Learning (ML), Random Forest Regression (RFR) (RFR), Renewable Energy.

TABLE OF CONTENTS

	Page
DECLARATION.....	ii
DEDICATION	iii
ACKNOWLEDGEMENT	iv
ABSTRACT.....	v
TABLE OF CONTENTS	vi
LIST OF FIGURES	ix
LIST OF TABLES	xii
LIST OF ABBREVIATIONS	xiii
CHAPTER 1	1
INTRODUCTION	1
1.1 Prolegomena.....	1
1.2 Objectives.....	2
1.3 Background and Motivation.....	2
1.4 Problem in Brief.....	3
1.5 Proposed Solution	3
1.6 Resource Requirement	4
1.7 Structure of the Thesis.....	4
1.8 Summary	4
CHAPTER 2	5
EVOLUTION OF MAXIMUM POWER POINT TRACKING SYSTEMS	5
2.1 Introduction	5
2.2 Gestation of Maximum Power Point Tracking Systems	5
2.3 Major Developments	23
2.4 Future Directions of MPPT Systems.....	24
2.5 Summary of Challenges in MPPT Systems	25
2.6 Problem Definition.....	26
2.7 Summary	26
CHAPTER 3	27
TECHNOLOGY ADOPTED.....	27

3.1	Introduction	27
3.2	Machine Learning Based MPPT Technology	27
3.3	Python.....	28
3.4	Sklearn Library.....	28
3.5	Random Forest Regression (RFR)	30
3.6	Simulation & Data Capturing Software	31
3.6.1	MATLAB	31
3.6.2	Simulink.....	32
3.6	Summary	33
CHAPTER 4		34
NOVEL APPROACH TO MAXIMUM POWER POINT TRACKING (VCIT-MPPT)		34
4.1	Introduction	34
4.2	Hypothesis.....	34
4.3	Input Process Output Model.....	34
4.4	Input	36
4.5	Output.....	36
4.6	Process.....	37
4.7	Features	38
4.8	Users.....	38
4.9	Summary	38
CHAPTER 5		40
DESIGNING OF VCIT-MPPT		40
5.1	Introduction	40
5.2	Top Level Architecture of VCIT-MPPT System	40
5.3	Modules in the Design.....	42
5.3.1	Solar Panel Interface Module	42
5.3.2	MPPT Control Module	43
5.3.3	ML Engine	45
5.4	Selection of ML Model	46
5.5	Model Description.....	47
5.6	Summary	48
CHAPTER 6		50

IMPLEMENTATION.....	50
6.1 Introduction	50
6.2 OS & Platforms	50
6.3 Installation of Required Software	51
6.3.1 Installation of Python 3.11.....	51
6.3.2 Installation of Jupyter Notebook 6.4.12	52
6.3.3 Installation of MATLAB R2023b and Simulink 23.2.....	53
6.4 Data Acquisition.....	54
6.3 Generating Dataset	56
6.4 Model Training Process	59
6.5 Predicting the Voltage Reference.....	62
6.5 System Interfaces	63
6.6 Summary	64
CHAPTER 7	65
EVALUATION OF VCIT-MPPT SYSTEM.....	65
7.1 Introduction	65
7.2 Selection of ML Model	65
7.3 Performance Evaluation	70
7.4 Dataset.....	72
7.5 Model Training.....	73
7.6 Testing of the Novel Design.....	74
7.7 Evaluation of the Novel Design	84
7.8 Results	87
7.9 Summary	87
CHAPTER 8	89
CONCLUSION AND FURTHER WORK	89
REFERENCES	91

LIST OF FIGURES

	Page
Figure 2.1: Annual power capacity expansion	6
Figure 2.2: V-I Characteristics of a Solar Panel	6
Figure 2.3: Typical MPPT Block Diagram	7
Figure 2.4: Basic Impedance Matching Circuit of MPPT	7
Figure 2.5 DC-DC converter for MPPT	8
Figure 2.6: Effects on MPP by Varying Temperature and Irradiance	8
Figure 2.7: Relationship between the OC voltage and temperature	9
Figure 2.8: Flow chart of hill climbing algorithm	10
Figure 2.9: PV curve for various irradiance levels	11
Figure 2.10: Flow chart of INC algorithm	12
Figure 2.11: Flow chart of OC voltage method	13
Figure 2.12: Flow chart of SC current method	14
Figure 2.13 : Two layer ANN in MPPT	15
Figure 2.14: ANN model for MPP voltage estimator.....	16
Figure 2.15: Learning & running stages of ML algorithms	19
Figure 4.1: IPO model of VCIT-MPPT system	35
Figure 4.2: Inputs & Outputs of VCIT-MPPT System	37
Figure 5.1: Top Level Architecture of VCIT-MPPT System	41
Figure 5.2: MALTAB Simulink solar panel interface module integrated to the PV panel ..	43
Figure 5.3: MPPT control module	44
Figure 5.4: ML Engine designed in MATLAB Simulink	46
Figure 6.1: Checking the Python version installed	51
Figure 6.2: Jupyter server	52
Figure 6.3: Jupyter notebook interface in a web browser	53
Figure 6.4: MATLAB R2023b	53
Figure 6.5: Simulink 23.2 installed on MATLAB R2023b	54
Figure 6.6: Conventional MPPT simulated in the MATLAB Simulink software simulator	55
Figure 6.7: MATLAB Simulink software simulator data logging mode	55
Figure 6.8: Conventional MPPT simulated in MATLAB Simulink software simulator	

with enabled data logging features	66
Figure 6.9: Output dataset displayed in MATLAB	57
Figure 6.10: Generating dataset table in MATLAB	57
Figure 6.11: MATLAB script to generate CSV dataset	58
Figure 6.12: CSV dataset for conventional MPPT using MATLAB	58
Figure 6.13: Importing of Python libraries features	59
Figure 6.14: Importing of dataset and converting to DataFrame	59
Figure 6.15: Checking for missing values and applying "ffill" to fill missing data	60
Figure 6.16: Splitting the MPPT dataset into training and testing	60
Figure 6.17: Training the VCIT-MPPT model using Random Forest Regression	61
Figure 6.18: Accuracy of the VCIT-MPPT model	61
Figure 6.19: Defining MPPT function in VCIT-MPPT	62
Figure 6.20: Designing of VCIT-MPPT system in MATLAB Simulink	63
Figure 6.21: Py_MPPT function block code	64
Figure 7.1: VCIT-MPPT dataset description using Jupyter Notebook	65
Figure 7.2: Visualization of VCIT-MPPT dataset using Seaborn pairplot() function	66
Figure 7.3: Correlation of VCIT-MPPT dataset using pandas corr() function	66
Figure 7.4: Visualization of VCIT-MPPT dataset using seaborn heatmap() function	67
Figure 7.5: Defining, training, evaluating and plotting various models	69
Figure 7.6: Selection of the machine learning (ML) model for the VCIT-MPPT system ..	69
Figure 7.7: Actual reference voltage values vs model predicted reference voltage values using Random Forest Regression	70
Figure 7.8: Variance and bias plot w.r.t. true line	71
Figure 7.9: Description of Dataset	72
Figure 7.10: Splitting the dataset into training and testing	73
Figure 7.11: Training the model using Random Forest Regression	74
Figure 7.12: Fluctuation of voltage reference in conventional MPPT at STC	75
Figure 7.13: Fluctuation of wattage in conventional MPPT at STC	75
Figure 7.14: Voltage reference in VCIT-MPPT at STC	76
Figure 7.15: Wattage in VCIT-MPPT at STC	76
Figure 7.16: Fluctuation of wattage in conventional MPPT at Temp. 10°C & Irr. Level	

1,020 W/m ²	77
Figure 7.17: Wattage in VCIT-MPPT at Temp.10°C & Irr. Level 1,020 W/m ²	77
Figure 7.18: Fluctuation of wattage in conventional MPPT at Temp. 15°C & Irr. Level 600 W/m ²	78
Figure 7.19: Wattage in VCIT-MPPT at Temp. 15°C & Irr. Level 600 W/m ²	78
Figure 7.20: Fluctuation of wattage in conventional MPPT at Temp. 20°C & Irr. Level 800 W/m ²	79
Figure 7.21: Wattage in VCIT-MPPT at Temp. 20°C & Irr. Level 800 W/m ²	79
Figure 7.22: Fluctuation of wattage in conventional MPPT at Temp. 22°C & Irr. Level 1,100 W/m ²	80
Figure 7.23: Wattage in VCIT-MPPT at Temp. 22°C & Irr. Level 1,100 W/m ²	80
Figure 7.24: Fluctuation of wattage in conventional MPPT at Temp. 25°C & Irr. Level 1,350 W/m ²	81
Figure 7.25: Wattage in VCIT-MPPT at Temp. 25°C & Irr. Level 1,350 W/m ²	81
Figure 7.26: Fluctuation of wattage in conventional MPPT at Temp. 30°C & Irr. Level 1,250 W/m ²	82
Figure 7.27: Wattage in VCIT-MPPT at Temp. 30°C & Irr. Level 1,250 W/m ²	82
Figure 7.28: Fluctuation of wattage in conventional MPPT at Temp. 35°C & Irr. Level 1,400 W/m ²	83
Figure 7.29: Wattage in VCIT-MPPT at Temp. 35°C & Irr. Level 1,400 W/m ²	83
Figure 7.30 PV array configuration and technical data	85
Figure 7.31: Accuracy of the VCIT-MPPT model	86
Figure 7.32: Cross Validation Scores of the VCIT-MPPT model	86

LIST OF TABLES

	Page
Table 2.1: Fuzzy Logic Rule Table.....	18
Table 2.2: Comparison of various MPPT techniques	22
Table 7.1: Summary of conventional MPPT vs VCIT-MPPT test results	84

LIST OF ABBREVIATIONS

Abbreviation	Description
CUR	Current
FFILL	Forward Fill
INC	Incremental Conductance
IPO	Input Process Output
IRR	Irradiance
MAE	Mean Absolute Error
MSE	Mean Square Error
MPP	Maximum Power Point
MPPT	Maximum Power Point Tracking
PIP	Preferred Installer Program
PV	Photovoltaic
PWM	Pulse Width Modulator
P&O	Perturb and Observe
REF	Reference
RFR	Random Forest Regression
RMSE	Root Mean Square Error
STC	Standard Test Conditions
TEMP	Temperature
TVC	Temperature Voltage Converter
VOL	Voltage

CHAPTER 1

INTRODUCTION

1.1 Prolegomena

Origin of the solar cell dated back to 1883 when the first solar power generation was invented by Charles Fritts. A solar system converts sunlight into electrical energy through photovoltaic (PV) panels and known as solar power generation. With the global revolution, the consumption of electricity has increased tremendously today. For nations that significantly rely on fossil fuels for their economies and power supply, this is a serious cause for concern. At the same time, there is a huge demand for electricity generation. To meet the ever-increasing demand for energy under the current circumstances, it is important that renewable energy resources be used to the fullest extent possible. There is a growing fear that the world's energy supplies could run out soon due to the rapid depletion of fossil fuel reserves. Today resources such as steam, nuclear, biomass, geothermal, wind, thermal and solar are used to meet the demand for electricity. One such sustainable energy source is solar energy, an endless supply that emits no greenhouse gases and is accessible anywhere in the world. Solar photovoltaics is a simple and extremely efficient method of capturing solar energy for power generation. At present solar power generation shows highest development of among all renewable resources [1] [2]. The typical commercial solar panel's efficiency is approximately 18% [3]. By continuously monitoring and modifying the operating point to meet the maximum power point at which the product of current and voltage (Wattage) is highest accessible from the solar panels, Maximum Power Point Tracking (MPPT) techniques maximize the power output of PV systems [4]. MPPT is a critical component in achieving the highest possible efficiency and power generation from solar installations. Major contributions and attractions of MPPT are increased energy harvesting, improved system efficiency, adaptation to varying environmental conditions, compatibility with different PV technologies [5] [6]. However, traditional Maximum Power Point Tracking (MPPT) techniques often struggle with fluctuating environmental conditions, such as varying solar irradiance and temperature.

This chapter is organized to present objectives, background and motivation, problem in brief, proposed solution, resource requirement, structure of the thesis.

1.2 Objectives

Following objectives have been identified to develop machine learning based Maximum Power Point Tracking (MPPT) of solar panel.

- Objective 1: Critical review of evolution of Maximum Power Point Tracking (MPPT) techniques.
- Objective 2: In depth study of technologies used in modern Maximum Power Point Tracking (MPPT) techniques.
- Objective 3: Design and implement an ML model that utilizes inputs of solar panel voltage, current, irradiance, and temperature to optimize the performance of solar panel.
- Objective 4: Evaluate the implemented solution.

1.3 Background and Motivation

Maximum Power Point Tracking (MPPT) techniques optimize the power output of PV systems by continuously tracking and adjusting the operating point to match the maximum power point at which the product of current and voltage (Wattage) is maximum available from the solar panels. The most basic MPPT methods use a DC–DC converter that is run at a duty cycle that, under standard test conditions (STCs), generates the pre-defined MPP voltage (V_{MPP}). However, they are unable to reach the MPP when not operating at STC by using this method and effects on MPP by varying irradiance and temperature [3] [7]. Temperature Gradient method is using two sensors to monitor voltage and temperature, and Temperature Gradient method is unable to reach the MPP in low irradiance conditions [3]. Perturb and Observe (P&O) and Incremental Conductance (INC), two hill climbing techniques, perform well in slowly changing

irradiance profiles but may not be able to accurately identify power points in highly variable irradiance profiles. [3]. Methods using open circuit voltage and short circuit current Measure the parameters of the PV panel on a regular basis, then use the short circuit current to determine the MPP and there is an interruption in energy harvesting during the time current (I_{sc}) and voltage (V_{oc}) measurements are obtaining resulting lower efficiency of the PV generation [3]. AI techniques based Artificial Neural Networks (ANN) and Fuzzy Logic Control (FLC) are most commonly used in the maximum power point tracking (MPPT) [7]. Common issue is ANN & FLC controllers are using inputs such as irradiance as a dependent parameter of the current of the solar panel in which energy harvesting is interrupted. Moreover, one major requirement for the ANN to function at its best without having a high training error rate is an accurate, standardized, and appropriate training set of data. [8].

1.4 Problem in Brief

Existing MPPT methods often fail to capture the entire relationships between voltage, current, irradiance and temperature parameters which directly affect the optimal operating point of a PV panel without disturbing the energy harvesting process, to deliver the maximum power output of the PV system. As a result of this situation overall efficiency and performance of the photo voltaic system may compromise.

1.5 Proposed Solution

This thesis presents developing machine learning based Maximum Power Point Tracking (MPPT) model that utilizes independent voltage, current, irradiance, and temperature sensing without interruption to energy harvesting process as well as to improve performance of Photo Voltaic (PV) systems. The projected solution involves leveraging the capabilities of machine learning models to learn the complex relationships between the input parameters (voltage, current, irradiance, and temperature) and the output (optimal operating point of the PV panel) from the analysis of collected data which enables accurate and efficient MPPT under varying environmental conditions without

disturbing the energy harvesting process, to deliver the maximum power output of the PV system.

1.6 Resource Requirement

The design requires following resources.

Hardware Resources - Computing infrastructure for model development, testing and simulation.

Software Resources

- Development environments (Eg:- Python)
- Machine learning libraries (Eg:- scikit-learn)
- Simulation and modeling tools (Eg:- MATLAB, Simulink)

Dataset – Own data collected from simulation environment w.r.t. solar panel voltage, current, irradiance of sunlight, and ambient temperature which need to be fed into the model.

1.7 Structure of the Thesis

The thesis has been structured under 8 chapters. The consequent chapters will describe the literature review, technology adopted, approach, design, implementation, evaluation, conclusion and further work.

1.8 Summary

This proposal presented a novel method of developing a machine learning based Maximum Power Point Tracking (MPPT) model that utilizes independent voltage, current, irradiance, and temperature sensing without interruption to energy harvesting and to enhance the performance of PV systems. The proposed method will be modeled and tested and compared with traditional MPPT techniques. This solution will overcome the drawbacks of the traditional MPPT techniques.

CHAPTER 2

EVOLUTION OF MAXIMUM POWER POINT TRACKING SYSTEMS

2.1 Introduction

In chapter 1 we have presented overall picture of this thesis covering problem definition, research problem and the essentials of the solutions. This chapter gives a comprehensive literature review in the area of MPPT. The literature review has been structured covering gestation of MPPTs, its developments and future directions. Here we also summarize the research challengers in MPPT and define our research problem.

2.2 Gestation of Maximum Power Point Tracking Systems

Origin of the solar cell dated back to 1883 when the first solar power generation was invented by Charles Fritts. A solar system converts sunlight into electrical energy through photovoltaic (PV) panels and known as solar power generation. With the global revolution, the consumption of electricity has increased tremendously today. At the same time, there is a huge demand for electricity generation. Today resources such as steam, nuclear, biomass, geothermal, wind, thermal and solar are used to meet the demand for electricity. At present solar power generation shows highest development of among all renewable resources [2] [9]. As per International Renewable Energy Agency the annual capacity installations over last ten years of time duration (2002 - 2022), tendency of the moving towards to renewable energy sources shows significant improvement. Figure 2.1 shows annual power capacity expansion for last ten years and there was only 15% renewable energy share in year 2002 and same was grown up to 85% in year 2022 which is a significant improvement over a decade. Additionally, as of the conclusion of 2022, the worldwide cumulative installed capacity of solar photovoltaic (PV) systems had reached 1,047 gigawatts (GW). Of this total, 191 GW were newly added in 2022 alone. Notably, the majority, approximately 59%, of these installations took place in the Asian region [9].

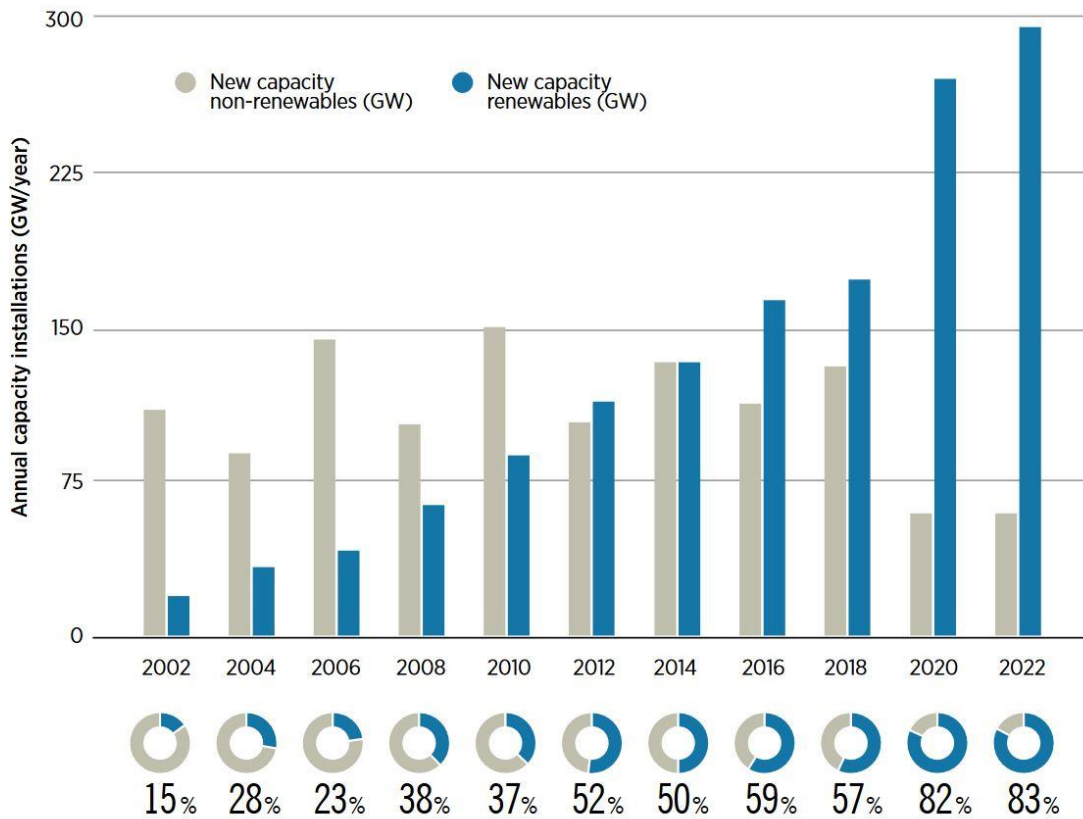


Figure 2.1: Annual power capacity expansion for last ten years [9].

The typical efficiency of a commercial solar panel stands around 18%. [3]. Techniques such as Maximum Power Point Tracking (MPPT) enhance the power output of photovoltaic (PV) systems by constantly monitoring and fine-tuning the operating point to align with the maximum power point. This point signifies the optimal combination of current and voltage, resulting in the highest wattage available from the solar panels. [4] [10]. VI characteristics of a solar panel is shown in Figure 2.2 [4].

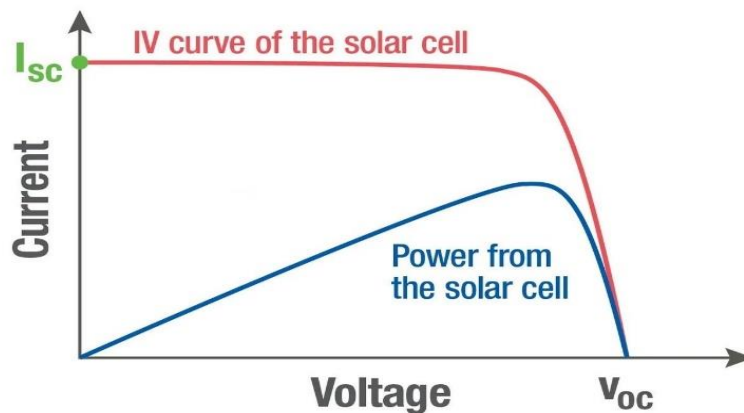


Figure 2.2: V-I characteristics of a solar panel [4].

Conventional MPPT techniques use current and voltage sensors as inputs to a microcontroller or the digital signal processor and vary the width of the pulse given to the gate signal of DC-DC converter. Typical MPPT block diagram is shown in Figure 2.3 [8] and the basic impedance matching circuit is shown in Figure 2.4 [11].

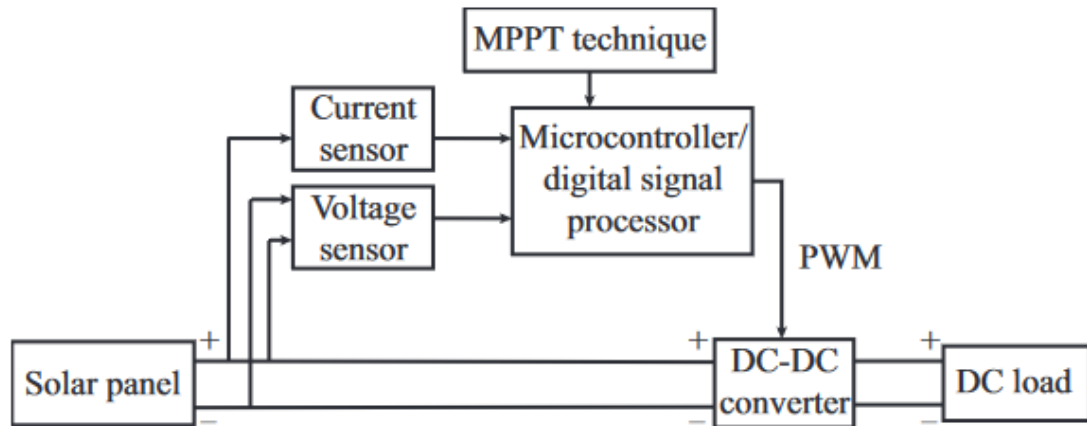


Figure 2.3: Typical MPPT block diagram [8].

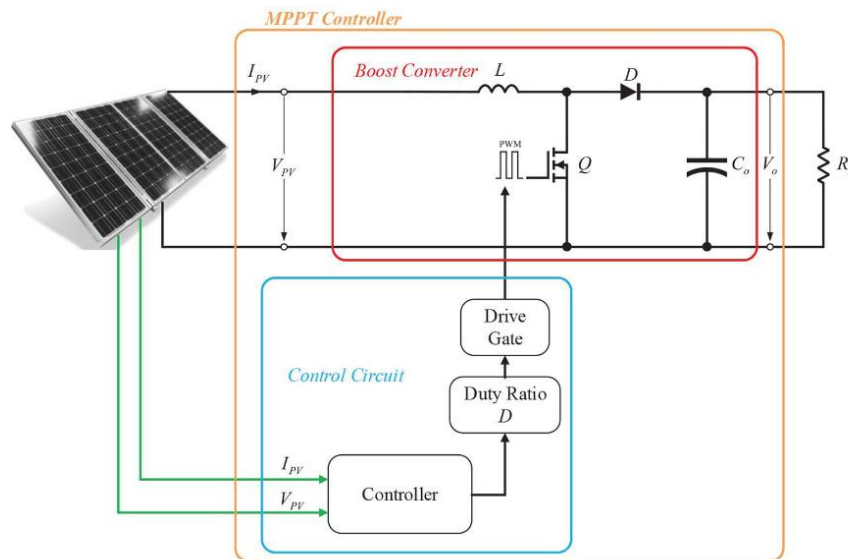


Figure 2.4: Basic impedance matching circuit of MPPT [11].

The DC-DC Converter Method represents one of the most straightforward MPPT techniques. It operates at a duty cycle configured to generate the predefined Maximum Power Point (MPP) voltage (V_{MPP}) corresponding to specific solar irradiance levels and temperatures. Figure 2.5 illustrates this basic MPPT setup, showcasing a DC-DC converter with a voltage gain denoted as G_v , which effectively simulates a variable resistance connecting the PV panel to the load side (R_L), alongside the duty cycle of the MPPT controller 'D' [7] [12].

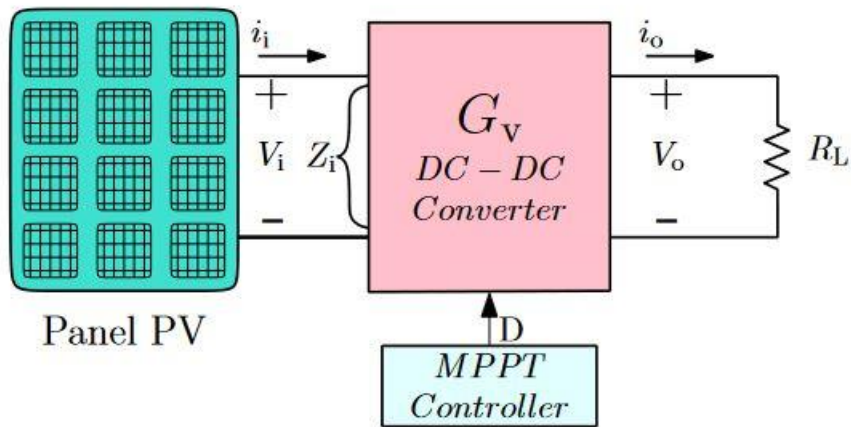


Figure 2.5 DC-DC converter for MPPT [12].

However, this method failed to attain the Maximum Power Point (MPP) outside of Standard Test Conditions (STC). The impact of varying irradiance and temperature on the MPP is illustrated in Figure 2.6 [7].

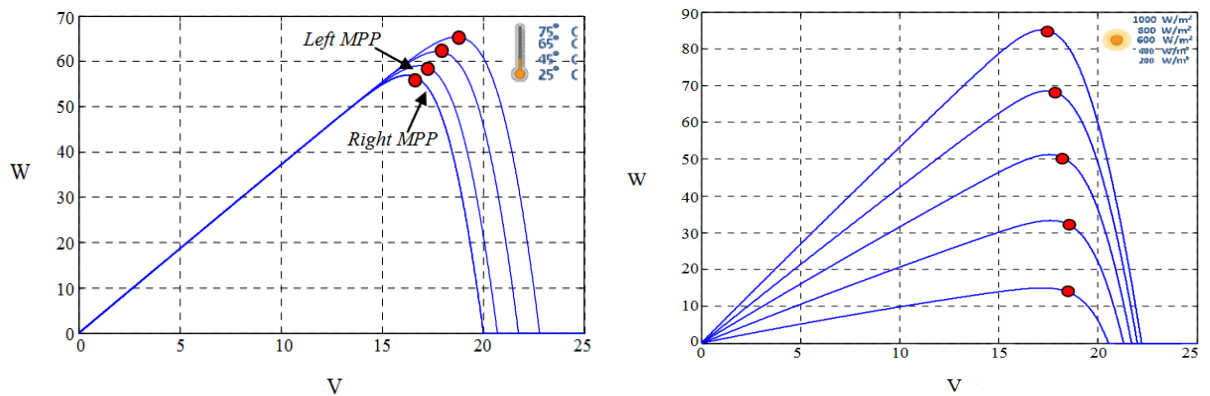


Figure 2.6: Effects on MPP by varying temperature and irradiance [7].

Temperature Gradient Method, Temperature is a crucial factor affecting the performance of photovoltaic (PV) systems. As the temperature of solar panels increases, their efficiency tends to decrease. Therefore, incorporating temperature considerations into MPPT algorithms is essential for optimizing the overall system performance [13]. The Temperature Gradient Method employs two sensors to monitor both the output voltage of the PV system and its temperature. Based on the PV module temperature, this method adjusts the VMPP accordingly [3] [14].

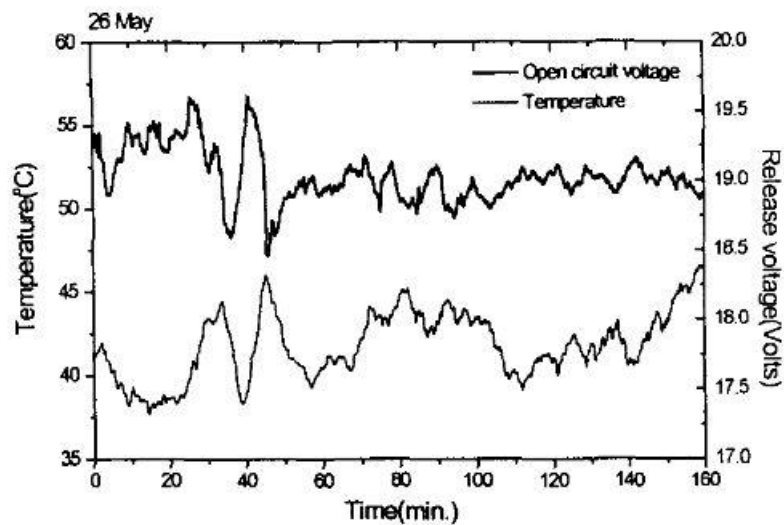


Figure 2.7: Relationship between the OC voltage and temperature [14].

Based on Figure 2.7, depicting the correlation between the open circuit voltage and the temperature of a PV cell, it becomes clear that the open circuit voltage increases in direct relation to the cell temperature [14]. The short circuit current of solar cells exhibits a weak dependency on temperature, typically showing a slight increase with rising temperatures. However, the short circuit current, which is directly related to the irradiance of the solar cell, was not factored into the temperature gradient method. Consequently, the temperature gradient method falls short in reaching the Maximum Power Point (MPP) under low irradiance conditions. [3] [14].

Hill Climbing Methods like Perturb and Observe (P&O) and Incremental Conductance (INC) yield favorable outcomes when applied to slowly changing irradiance profiles. They function as local search algorithms, persistently observing and shifting towards higher power outputs to reach the peak power point. The process concludes once it arrives at a peak value where no adjacent point exhibits a superior output [15]. Hence these methods and may fail to provide accurate power point tracking under high varying irradiance profiles and fast changing environmental conditions [3] [16] [15] [17].

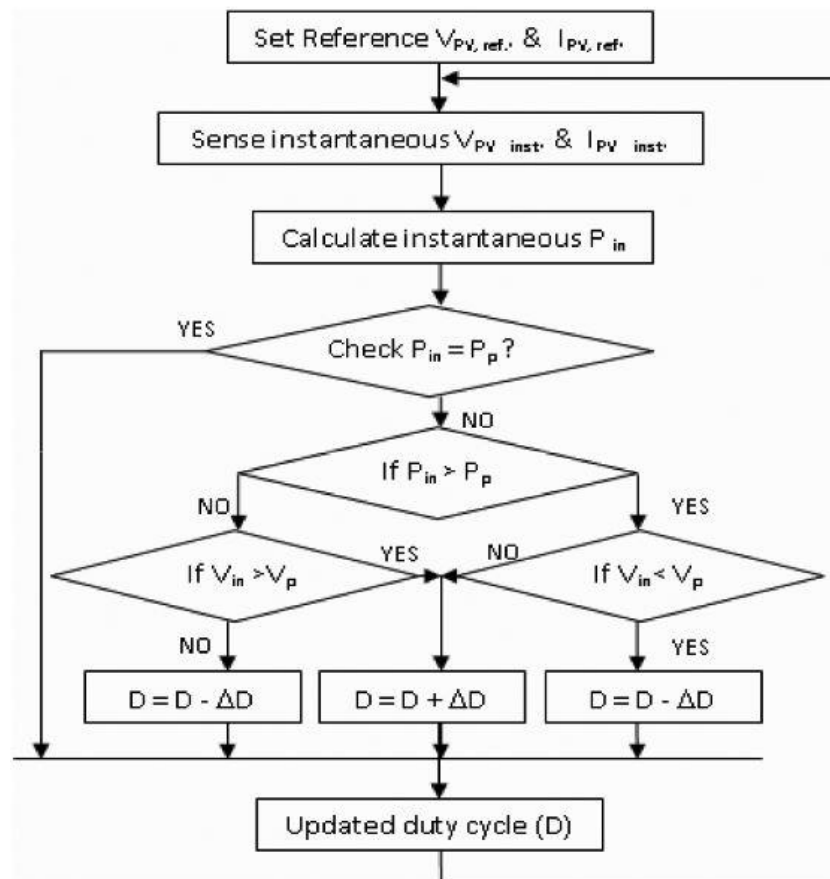


Figure 2.8: Flow chart of hill climbing algorithm [15].

The Incremental Conductance Algorithm is based on derivative (dP/dV) of the output power (P_{pv}) w.r.t. panel voltage (V_{pv}) will become zero at MPP. Figure 2.9 is showing the PV curve for selected 0.5 kW/m^2 , 0.75 kW/m^2 , 1.0 kW/m^2 irradiance levels.

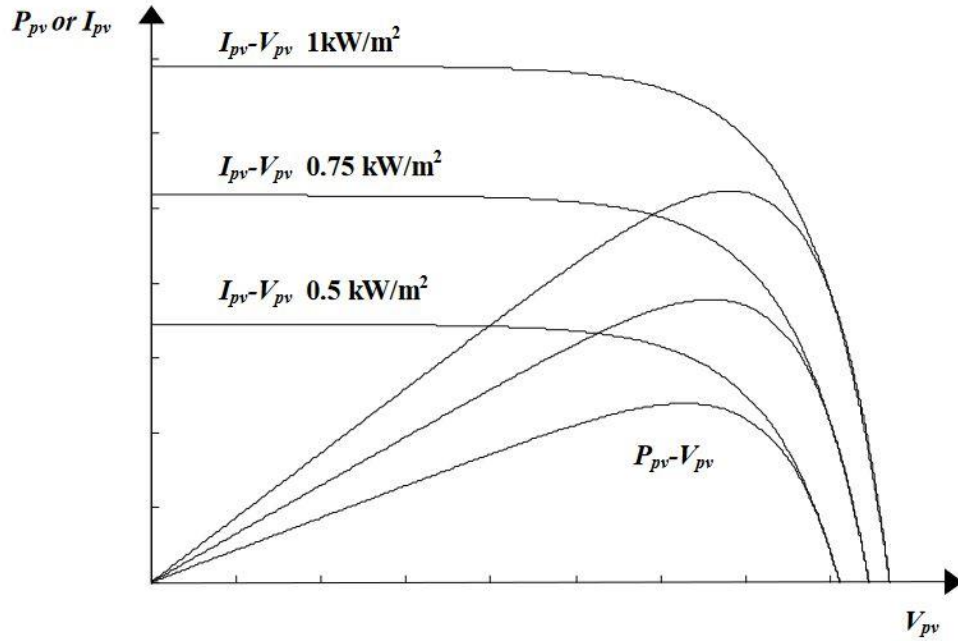


Figure 2.9: PV curve for various irradiance levels [17]

Hence,

$$\frac{dP_{pv}}{dV_{pv}} = \frac{d(I_{pv}V_{pv})}{dV_{pv}} = I_{pv} + V_{pv} \frac{dI_{pv}}{dV_{pv}} = 0 \quad \text{----- at MPP}$$

$$\frac{dP_{pv}}{dV_{pv}} = \frac{d(I_{pv}V_{pv})}{dV_{pv}} = I_{pv} + V_{pv} \frac{dI_{pv}}{dV_{pv}} > 0 \quad \text{----- at the left to MPP}$$

$$\frac{dP_{pv}}{dV_{pv}} = \frac{d(I_{pv}V_{pv})}{dV_{pv}} = I_{pv} + V_{pv} \frac{dI_{pv}}{dV_{pv}} < 0 \quad \text{----- at the right to MPP}$$

So same can be written as follows: -

$$\frac{dI_{pv}}{dV_{pv}} = -\frac{I_{pv}}{V_{pv}} \quad \text{----- at MPP}$$

$$\frac{dI_{pv}}{dV_{pv}} > -\frac{I_{pv}}{V_{pv}} \quad \text{----- at the left to MPP}$$

$$\frac{dI_{pv}}{dV_{pv}} < -\frac{I_{pv}}{V_{pv}} \quad \text{----- at the right to MPP}$$

So these equations were used to control the operating point of the PV panel using incremental and instantaneous conductance (dI_{pv}/dV_{pv} and I_{pv}/V_{pv}) [17] [18].

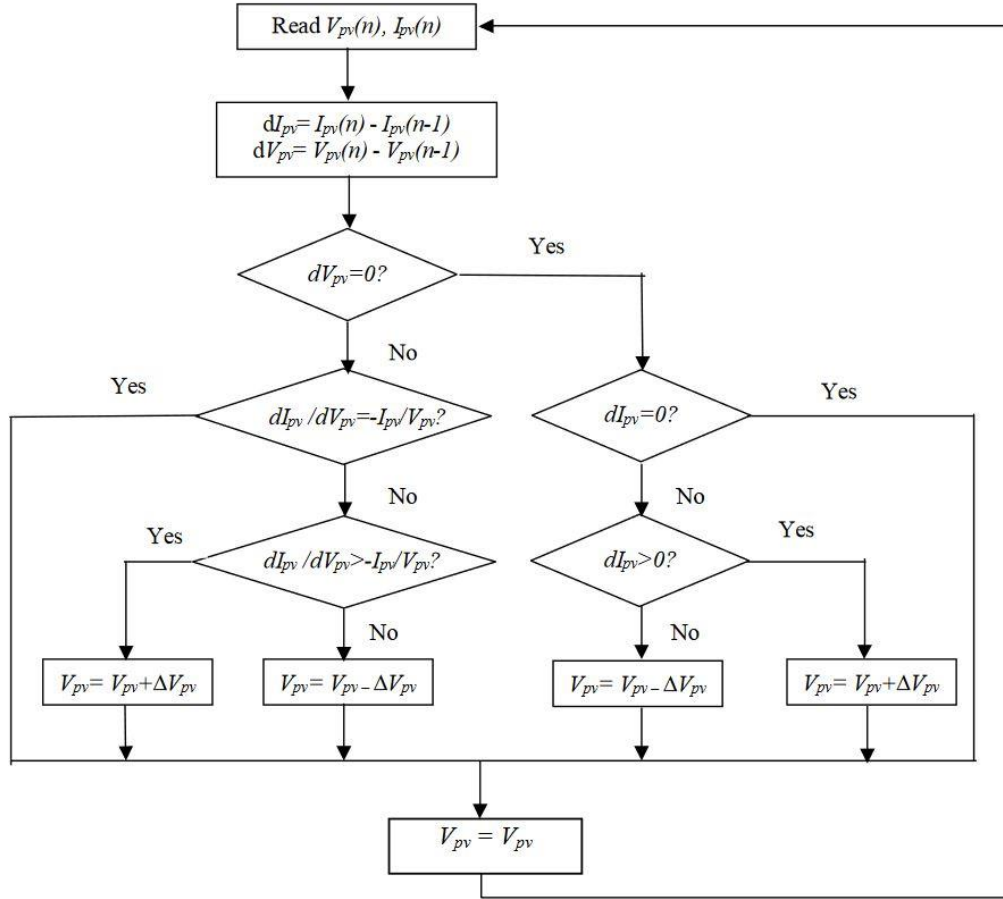


Figure 2.10: Flow chart of INC algorithm [17].

The Short Circuit Current & Open Circuit Voltage Methods conduct regular assessments of PV panel parameters, determining the Maximum Power Point (MPP) using the short circuit current, which correlates with the irradiation level, as well as the open circuit voltage, reference current under short circuit conditions, and reference voltage under open circuit conditions. The Short circuit Current method, the MPPT algorithm operates by varying the op. point of the solar system to ensure that its generated current is maximized under varying environmental conditions. This is achieved by continuously monitoring the short circuit current of the PV system.

The Open circuit Voltage method aims to optimize the operating point of the solar array by maximizing the open circuit voltage across varying environmental conditions. This approach entails ongoing monitoring and adjustments to ensure the solar system is operating at the point where, open circuit voltage is optimized.

In both the short circuit current & open circuit voltage methods, the percentage values derived from the PV panel's short circuit current & open circuit voltage are utilized in calculations to align with MPP of the solar panel. However, there is disruption in energy harvesting during the periods when short circuit current (I_{sc}) & open circuit voltage (V_{oc}) measurements are being obtained, resulting in decreased efficiency of PV generation. These methods demonstrate optimal performance under stable environmental conditions of irradiance and temperature [3] [6] [19] [20].

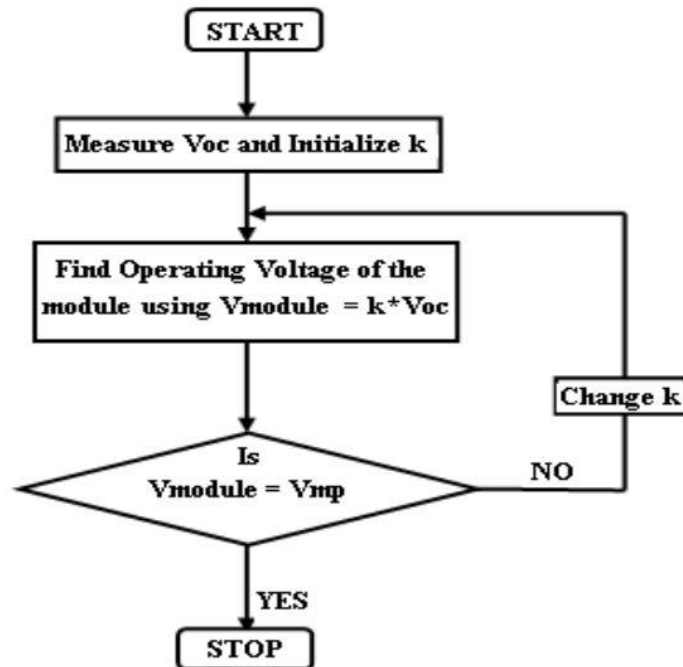


Figure 2.11 : Flow chart of OC voltage method [19].

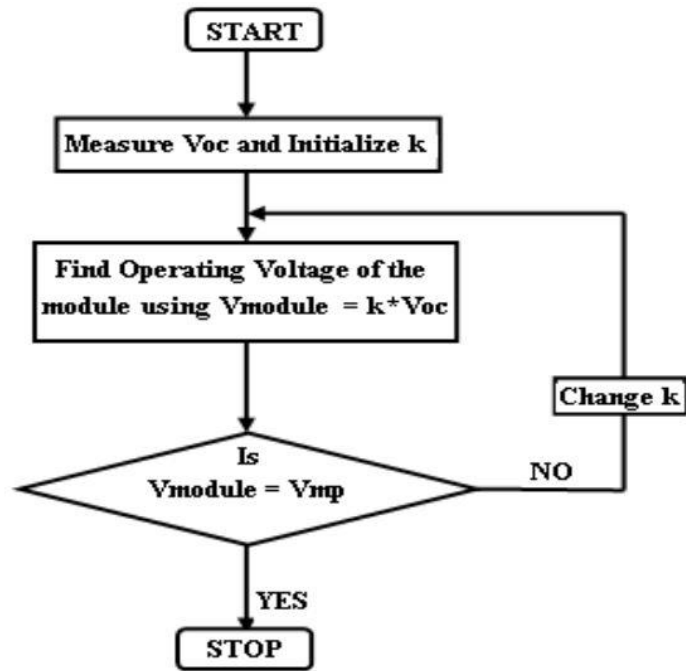


Figure 2.12 : Flow chart of SC current method [20].

In open circuit voltage method:-

$$V_{MPP} = k * V_{OC}$$

the value of the constant 'k' is varying from 0.7 – 0.8 depending upon the characteristics of the PV panel [19].

On the other hand in short circuit current method:-

$$I_{MPP} = k * I_{SC}$$

the value of the constant 'k' is approximately around 0.9 depending upon the characteristics of the PV panel [20].

Artificial Neural Networks (ANNs) are used in MPPT algorithms to learn the non-linear relations between I/P variables & the corresponding optimal voltage and current values. Artificial Neural Networks (ANNs) have been increasingly utilized in Maximum Power Point Tracking (MPPT) algorithms for photovoltaic (PV) systems due to their ability to improve tracking accuracy and adaptability to changing environmental conditions [5] [7]. ANNs are computational models that draw inspiration from the structure and functionality of biological neural networks in brain [5] [21] [22]. ANNs are made to link different parameters to specific data points without requiring intricate mathematical bases or calculations. A method called as supervised learning is used to train ANNs. This method uses datasets made up of values for input-output parameters to train the network. A training dataset and a validation dataset are commonly divided into two groups. The network is trained using the training dataset, and its performance is evaluated using the validation dataset. Many distinct neurons comprise an artificial neural network (ANN), and each neuron is connected to the others by a fractional number known as weight. These weights are adjusted during the training in order to exactly predict the results. When the error goes down to a permissible threshold, the weights continue to be constant [22]. Basic two-layer architecture taking input as PV panel voltage and current to obtain MPP voltage as output is depicted in Figure 2.13 is the most basic two layer ANN in MPPT [8] but it fails to capture the direct temperature and irradiance changes in the environment.

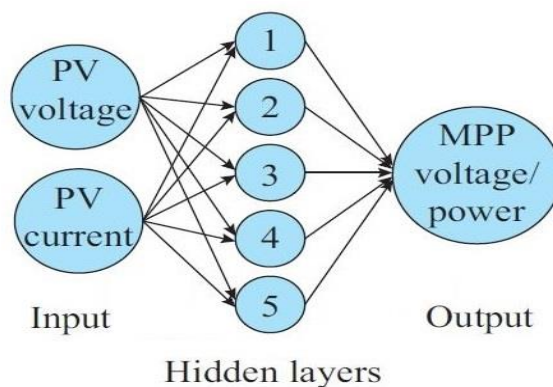


Figure 2.13 : Two layer ANN in MPPT [8].

In the context of MPPT, ANNs are employed to enhance the performance of traditional algorithms by providing adaptive and intelligent decision-making capabilities [21]. Various ANN models have been developed in recent past for MPPT. Most common ANN model uses irradiance (G), temperature (T) as the inputs to the ANN and determines the output voltage which is the V_{MPP} and in some models it uses same inputs but obtaining both output voltage (V_{MPP}) and output current (I_{MPP}) which makes a complex model [23]. Figure 2.14 displays the ANN model for MPP voltage estimator [24]. This MPP voltage is given to the PWM generator.

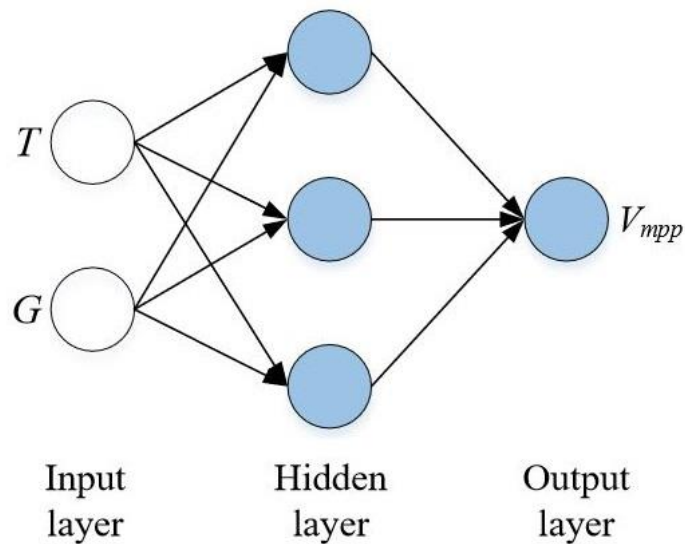


Figure 2.14: ANN model for MPP voltage estimator [24].

However, currently ANNs are extensively employed for MPPT owing to their capacity to grasp the intricate nonlinear and dynamic characteristics of PV systems [8]. Moreover, application of ANN to replace conventional MPPT techniques have improved the accuracy & adaptability of ensuring the MPP is maintained across various operating conditions [21] [22].

Although ANNs have demonstrated enhanced efficiency and accuracy in MPPT systems, a key drawback of ANN based MPPT methods are the necessity for a substantial volume

of data to effectively train the network. This requirement could pose a significant hurdle for certain applications, particularly those operating in remote areas or with limited data collection capabilities. Additionally, the availability of accurate, standardized, and comprehensive training datasets remains a primary limitation for ANNs to achieve optimal performance without encountering high training errors [8]. Furthermore, the training process could be computationally and time-consuming, which could be an issue for real-time applications [24] [25].

Fuzzy Logic Controller agents can be incorporated to implement a variable step size perturbation and to autonomously adjust the steps based on variations in irradiance and temperature factors. This integration addresses limitations in traditional MPPT techniques like P&O, which fail to adapt perturbation steps in response to rapid changes in weather conditions such as irradiation or temperature [26]. Fuzzy Logic Controllers are a type of control system that utilizes fuzzy logic to handle imprecision and uncertainty in complex systems. In the framework of MPPT for solar systems, Fuzzy Logic Controllers can be employed to dynamically adjust the operating point of the solar array based on real-time environmental conditions. FLC is a MPP control system based on fuzzy logic that converts analogue I/Ps into fuzzy logical values between zero & one, presenting the idea of fractional truth, in which the variable value may be either completely true or totally false. Fuzzy trackers keep an eye on the MPP even in the event of erroneous inputs and that is the reason FLCs are considered intelligent. Fuzzy controllers operate without the necessity of a mathematical model. The typical process of fuzzy control involves three main steps: fuzzification, rule-based lookup tables & de-fuzzification. During the initial modeling phase, numerical input variables are transformed into linguistic variables using a membership function with five fuzzy levels: NB (Negative Big), NS (Negative Small), ZE (Zero), PS (Positive Small), and PB (Positive Big). In an MPPT fuzzy logic controller, the inputs typically consist of an Error (E) and a change in error (ΔE), which are computed as follows: -

$$E(i) = \frac{P_{pv}(i) - P_{pv}(i-1)}{V_{pv}(i) - V_{pv}(i-1)}$$

$$\Delta E(i) = E(i) - E(i - 1)$$

Where:

E – Error

P_{PV} – Power of the PV Panel

V_{PV} – Voltage of the PV Panel

ΔE – Change in Error

Once ΔE and E have been transformed into linguistic variables and processed according to the Fuzzy Logic Rule Table depicted in Table 2.1, the FLC's output, denoted as ΔD (change in duty-cycle) of the power converter, can be determined. During the defuzzification phase, the linguistic output of the FLC is converted into a numerical variable, generating an analog signal that drives the power converter towards the MPP. The FLC MPPT demonstrates effective performance under changing weather conditions.

However, establishing the rule base table and selecting the appropriate error computation method are crucial for its effectiveness. Notably, mathematical modeling is unnecessary when employing this approach for MPPT. Moreover, the FLC system's stability around the maximum power point is improved due to reduced fluctuations. Nonetheless, challenges arise in fine-tuning the control rules, scale factor, and membership function [5] [26].

$\Delta(E)$ Error (E)	Change in Error				
	NB	NS	ZE	PS	PB
NB	ZE	ZE	NB	NB	NB
NS	ZE	ZE	NS	NS	NS
ZE	NS	ZE	ZE	ZE	PS
PS	PS	PS	PS	ZE	ZE
PB	PB	PB	PB	ZE	ZE

Table 2.1: Fuzzy Logic Rule Table [5].

Regression involves establishing a mathematical relationship between multiple independent variables and a dependent variable. In case of MPPT in solar panels the independent variables include environmental factors like irradiance & temperature, which significantly effect the performance of PV modules. Hence, a predictive model that correlates these variables with the optimal operating point, enabling real-time adjustments for to capture maximum energy / power output of the solar panel. The model takes into account the complex interplay between irradiance, temperature, and the resulting electrical characteristics of the PV modules. This enables to dynamically respond to changing conditions, ensuring that the system operates efficiently across a wide range of scenarios. Regression which comes under machine learning. The machine learning model must go through two steps in order to be completely functional. Those are the testing stage & learning stage, also known as the training stage. First model required to be trained on historical data then based on the learnings of the trained data model can be validated using testing data. In regression based MPPT models certain potion of data set of irradiances, voltage are given as input data and reference voltage (V_{ref}) is given as the output data [27].

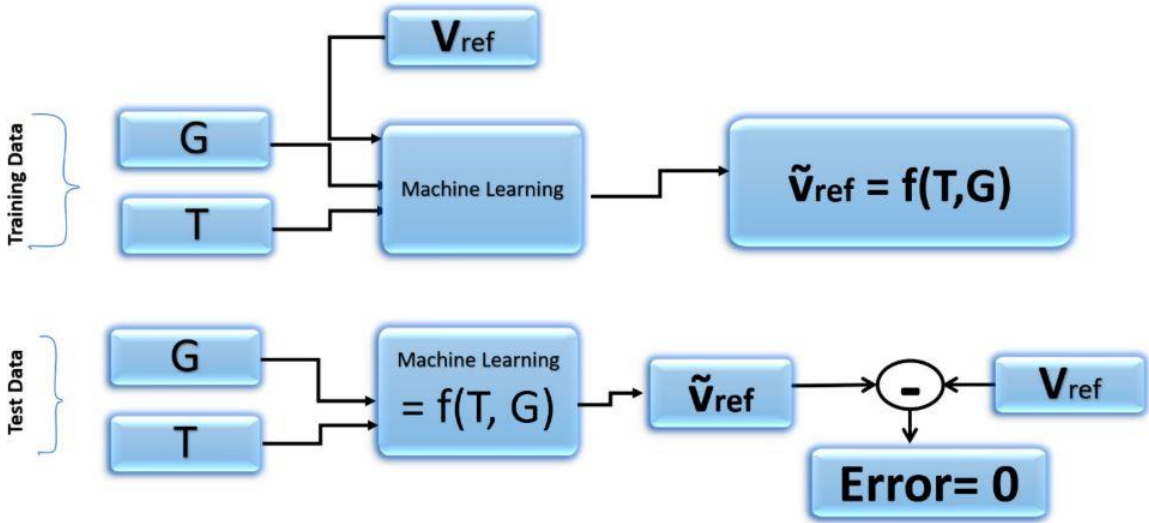


Figure 2.15: Learning & running stages of ML algorithms [27].

The reference voltage (V_{ref}) generator block accurately determines the reference voltage by leveraging historical data on environmental factors like irradiance & temperature. Through training, it predicts an optimal V_{ref} corresponding to the MPP of the solar panel for specific temperature-irradiation combinations. To ensure functionality, the machine learning block undergoes two phases: learning (training) and running (testing).

During learning stage, the aim is to develop and validate a function that correlates temperature and irradiation with V_{ref} using historical data. Then resulting model from this phase is utilized in the running phase to forecast V_{ref} value for new testing data, unseen during the learning phase.

These phases are depicted in Figure 2.15. Accuracy of the ML model's prediction is assessed by means of Root Mean Square Error (RMSE), represented as

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (V_{ref} - \tilde{V}_{ref})^2}$$

Regression-based MPPT is its adaptability to varying environmental conditions. The algorithm can continuously learn and optimize its predictions based on real-world performance. Further, leveraging historical data allows regression-based models to make accurate predictions, enabling the system to adapt to changing conditions and improve overall efficiency. Furthermore, regression-based models can provide more accurate predictions of the optimal operating point by considering multiple environmental parameters simultaneously resulting enhancing overall system efficiency comparatively in less complexity when compared with ANNs. One of the major challenge in Regression-based MPPT is also the need for substantial training data to train the model [28] [29]. Moreover, regression-based MPPT proposed as of now are only using basic temperature and irradiance as their parameters [28] [29] [30].

Further, comparison of various MPPT techniques with their advantages and limitations is tabulated in below table:-

MPPT Technique	Advantages	Limitations
DC-DC Converter Method	Simple implementation. Cost effectiveness. Suitable for fixed MPP voltage applications.	Limited adaptability to changing environmental conditions. Inability to track varying MPPs.
Temperature Gradient Method	Utilizes temperature differences to estimate MPP. Low complexity. Suitable for temperature-varying environments.	Requires accurate temperature measurement. Limited applicability in non-uniform temperature profiles. Unable to reach the maximum power in low irradiance conditions.
Perturb and Observe (P&O)	Simple, easy to implement. Suitable for slow-varying irradiance conditions. Low computational requirement.	Oscillations around the MPP under dynamic conditions and leading to energy losses. Slow tracking response and less effective under rapidly changing environmental conditions.
Incremental Conductance	Improved tracking efficiency over P&O. Quick response to slow varying irradiation levels.	Complex algorithm requiring precise current and voltage measurements. Sensitive to measurement noise. Inability to provide accurate power point tracking under high varying irradiance profiles.
Short Circuit Current and Open Circuit Voltage Methods	Simple and easy to implement.	Limited accuracy due to non-linear characteristics of PV panels.

	<p>Low cost MPPT system.</p> <p>Low computational requirements.</p> <p>Good performance in steady irradiance and temperature conditions.</p>	<p>Less effective in rapidly changing environmental conditions.</p> <p>Interruptions during measurements lead to lower efficiency.</p>
Artificial Neural Networks (ANN).	<p>Adaptive and self-learning capability.</p> <p>Can handle non-linear characteristics of PV systems.</p> <p>Adaptable to changing environmental conditions.</p>	<p>Complex training process and computational requirements.</p> <p>Requires a large amount of training data.</p> <p>Computationally and time-consuming training process.</p> <p>Limited availability in accurate, standardized and proper training set of data.</p>
Fuzzy Logic Control (FLC).	<p>Handles imprecision and uncertainty in the system.</p> <p>Improved tracking performance under dynamic conditions.</p>	<p>Complex design, fine-tuning the control rules, scale factor, and membership function.</p>
Regression-based MPPT	<p>Adaptable to varying environmental conditions.</p> <p>More accurate predictions by considering multiple parameters.</p>	<p>Requires substantial training data.</p> <p>Limited availability in accurate, standardized and proper training set of data.</p>

Table 2.2: Comparison of various MPPT techniques

Moreover, MPPT is a critical component in achieving the highest possible efficiency and power generation from solar installations [5] [6]. Major contributions and attractions of maximum power point tracking are:-

Enhanced Energy Capture: Through MPPT methods, solar panels are optimized to function at their peak power performance, known as the maximum power point (MPP). By constantly monitoring and fine-tuning the operational parameters, MPPT guarantees that solar panels consistently operate at their MPP, thus maximizing the extraction of energy from sunlight.

Improved System Efficiency: MPPT techniques optimize the conversion of solar energy into usable electrical power. By dynamically adjusting the operating conditions of the PV system, MPPT reduces power losses and improves overall system efficiency. This translates into higher energy yields and increased return on investment for solar power installations.

Adaptation to Varying Environmental Conditions: Solar panel performance is affected by factors such as temperature, shading, and changing sunlight intensity. MPPT techniques continuously monitor these environmental conditions and adjust the operating parameters of the PV system accordingly. This adaptive capability ensures that the solar panels always operate at their optimum regardless of the prevailing conditions, resulting in maximum power generation.

Compatibility with Different PV Technologies: MPPT is applicable to various types of solar panel technologies, including crystalline silicon, thin-film, and emerging technologies. Its adaptive nature allows it to work effectively with different panel characteristics, ensuring efficient power extraction across a wide range of PV systems.

2.3 Major Developments

Improved Algorithms: Researchers have developed more sophisticated MPPT algorithms to enhance tracking accuracy and efficiency. These algorithms include Perturb and

Observe (P&O), Incremental Conductance (INC), and Short Circuit Current & Open Circuit Voltage methods. They address challenges such as noise, partial shading, and fast-changing environmental conditions, resulting in better performance and higher energy yields [16]. Hardware and Converter Topology Advances: Innovations in power electronics and converter topologies have contributed to the expansion of much more efficient and compact MPPTs. Integration of advanced power semiconductor devices, such as wide-bandgap materials (Eg:- Silicon carbide [SiC] and Gallium Nitride [GaN]), and the use of multi-level and interleaved converter configurations have improved overall system efficiency and reduced losses [31]. Adaptive MPPT techniques dynamically adjust control parameters based on changing environmental conditions, enabling optimal power extraction. Hybrid MPPT methods combine multiple algorithms to leverage their individual strengths and improve overall tracking efficiency. These developments provide enhanced adaptability and robustness to MPPT systems [32]. AI techniques are being employed in MPPT systems to optimize power generation by leveraging historical and real-time data. These approaches enable intelligent decision-making, accurate modeling of nonlinear characteristics, and prediction of optimal operating points and enhancing system performance [33].

2.4 Future Directions of MPPT Systems

Future directions of MPPT systems involve ongoing research and development efforts to further improve their performance and adaptability. The incorporation of MPPT systems with energy storing technologies, such as gel batteries or supercapacitors, can empower better application of harvested energy. MPPT techniques can be enhanced to consider energy storage levels and optimize power flow between the PV system and storage devices, leading to improved self-consumption and grid interaction [34]. Smart Grid Integration: MPPT systems can be integrated into smart grid frameworks to enable bidirectional power flow, demand response, and grid stability enhancement. By utilizing communication and control technologies, MPPT systems can participate in grid-level optimization and support grid services, such as voltage regulation and frequency control [35]. Hybrid and Multi-Source MPPT: The development of MPPT techniques for hybrid

and multi-source renewable energy systems, such as combining solar & wind sources, is an ongoing research area. The objective is to design MPPT systems that can simultaneously track the maximum power points of multiple sources and optimize power extraction from each source in a coordinated manner [36].

2.5 Summary of Challenges in MPPT Systems

The most basic MPPT methods involve employing a DC–DC converter set to a specific duty cycle, resulting in the desired maximum power point (MPP) voltage (VMPP) under standard test conditions (STCs). Nonetheless, these techniques face difficulty in adjusting to variations in environmental conditions outside of STCs, thus hindering their ability to reach the MPP [7] [12]. The Temperature Gradient method employs a pair of sensors to oversee both the voltage and temperature of a photovoltaic system. However, it struggles to attain the Maximum Power Point (MPP) because it is constrained by its inability to adapt well to irregular temperature distributions and low light intensity situations [3] [13] [14]. Hill climbing techniques like Perturb and Observe (P&O) and Incremental Conductance (INC) demonstrate effectiveness in steady irradiance changes but might struggle to precisely and efficiently track the MPP in quick fluctuating irradiance scenarios. This is because they tend to oscillate around the maximum power point during dynamic conditions, resulting losses in energy [3] [16] [15] [17]. The short circuit current & open circuit voltage techniques periodically measure parameters of PV panels and determine the Maximum Power Point (MPP) using the short circuit current. However, energy harvesting is interrupted during these measurements, leading to decreased efficiency in PV generation. Moreover, this method is less efficient in quickly changing environmental conditions [3] [6] [19] [20]. AI techniques based Artificial Neural Networks (ANN) and Fuzzy Logic Control (FLC) major AI techniques used in the MPPT [7] [21] [22]. A major constraint for artificial neural networks (ANNs) is the availability of a precise, uniform, and adequate training dataset. Moreover, the training process demands intricate computational resources and consumes considerable time [8] [21] [22] [24] [25]. In case of Fuzzy Logic Control (FLC) approach poses challenges when it comes to fine-tuning the control rules, scale factor, and membership function [5]

[26]. Regression-based MPPT also the need for substantial training data to train the model [28] [29]. However, regression-based MPPT proposed as of now are only using basic temperature and irradiance as their parameters which can be further improved by introducing the panel voltage and current as inputs the model.

2.6 Problem Definition

Based on the literature, it was identified existing MPPT methods often fail to capture the entire relationships between voltage, current, irradiance and temperature parameters which directly affect the optimal operating point of a PV panel without disturbing the energy harvesting process, to deliver maximum power output (wattage) of the solar system. As a result, overall efficiency and performance of the photo voltaic system may compromise.

2.7 Summary

This chapter presented the literature review in the area MPPT by covering thirty six well cited research papers. Chapter delves into the evolution of MPPT systems, providing a inclusive review of historical developments and current techniques. It outlines conventional MPPT methods like DC-DC converters, temperature gradient, and hill climbing methods, advanced machine learning approaches such as Artificial Neural Networks (ANNs) and Fuzzy Logic Controllers & using of regression methods in the field of MPPT. Research problem is defended as, inability of existing MPPT methods to capture the entire relationships between voltage, current, irradiance & temperature parameters which straightly affect the optimal operating point of a solar panel without disturbing the energy harvesting process, to deliver the maximum power output (wattage) of the PV system. Further, the chapter discusses recent advancements, including improved algorithms, hardware innovations, and the application of artificial intelligence. Future directions are explored, such as integration with energy storage and smart grids. Challenges in existing MPPT methods, like adaptability and the need for substantial training data, are highlighted. The chapter 3 presents our in-depth study of machine learning techniques by highlighting its importance to the area of MPPT.

CHAPTER 3

TECHNOLOGY ADOPTED

3.1 Introduction

Amongst the technologies that can be adopted to design machine learning based MPPT, as summarized in chapter two, regression-based model is identified to be more convenient considering the low computational requirements, low cost implementation and easy configuration to existing solar panels to replace conventional MPPT approaches. In this chapter, the focus is on the technologies and tools that have been employed in the development of the proposed regression-based model in machine learning as the technology adopted for developing machine learning based MPPT. Specifically, this chapter will cover the hardware and software used in machine learning based MPPT data capturing, programming languages and other development libraries that were utilized to create and assess the research project. These technologies were carefully chosen based on their effectiveness, ease of use, and compatibility with the methodologies used in this study. Each technology is briefly introduced and its respective role in this research project is explained.

3.2 Machine Learning Based MPPT Technology

In this approach regression-based machine learning model is trained by providing the inputs as instantaneous voltage of the solar panel via voltage sensor & measuring of instantaneous current drawn from the solar panel via a current transformer, measuring irradiance of the incident wave via an irradiance sensor, and temperature of the solar panel (ambient temperature) via temperature sensor. Then inputs the irradiance sensor and temperature sensor, PV panel voltage and the current transformer will be fed in to the regression-based machine learning model as a captured dataset.

One of the challenging parts during this research project was to obtain a dataset for training the machine learning model. No reliable datasets found to train the machine

learning model which consisting instantaneous voltage and instantaneous current drawn from the solar panel, irradiance of incident wave & ambient temperature (temperature of solar panel). Hence, using MATLAB and Simulink software said data was captured and exported as an CSV (comma separated values) file which can be used to train the machine learning model.

3.3 Python

Python was mainly used as the primary programming language for developing the regression-based MPPT model a due to its flexibility, ease of use, and extensive support for ML and the libraries. Python provides a large number of libraries and tools for data processing, analysis, and visualization, which were utilized in developing the system. More importantly, it's simple syntax and powerful capabilities make it an ideal choice for implementing different Artificial Intelligence (AI) related model and algorithms. Additionally, the language has a large community and a wide range of resources available, making it easier to troubleshoot and optimize the implementation.

Moreover, Python is a platform-independent language, that code written in Python can run on different operating systems / platforms such as MATLAB and Simulink without modification. This portability is advantageous for deploying MPPT algorithms across diverse hardware platforms. Further, Python provides robust support for interfacing with hardware components through libraries like PySerial and PyUSB. This capability allows to interact seamlessly with sensors and communication interfaces necessary for real-time monitoring and control of systems.

3.4 Sklearn Library

Scikit-learn, often referred to as sklearn, stands out as a highly favored and extensively utilized Python library for machine learning. It offers a user-friendly and effective solution for tasks in data mining and analysis. Leveraging foundational Python libraries

like NumPy, SciPy, and matplotlib, scikit-learn seamlessly integrates with the broader Python data science environment.

Scikit-learn offers a simple and consistent API, which makes it easy to use and learn. This library consists with an uniform approach across various algorithms and functionalities, allowing users to focus more on solving problems rather than dealing with implementation details.

The library offers an extensive array of both supervised and unsupervised learning methods, covering classification, regression, clustering, dimensionality reduction, model selection, and preprocessing approaches. Among the well-known algorithms supported by scikit-learn are linear regression, support vector machines (SVM), random forests regression, gradient boosting & numerous others.

Moreover, the library provides comprehensive resources for assessing model performance, including evaluation, validation, and metrics. It offers techniques like k-fold cross-validation and stratified k-fold cross-validation to gauge the generalizability of machine learning models. Additionally, scikit-learn includes functions to calculate diverse evaluation metrics such as accuracy, precision, recall and mean squared error.

This library includes a wide range of feature extraction and preprocessing techniques to handle different types of data. It provides utilities for feature scaling, normalization, encoding categorical variables, imputation of missing values, text processing, and feature selection. These preprocessing steps are crucial for improving the performance and robustness of machine learning models.

Moreover, Scikit-learn can be flawlessly incorporate with other popular Python libraries used in the data science ecosystem, such as pandas for data manipulation, seaborn for data visualization. This interoperability allows users to combine the strengths of different libraries within their machine learning workflows.

3.5 Random Forest Regression (RFR)

Selecting Random Forest Regression (RFR) for Maximum Power Point Tracking (MPPT) in solar energy systems presents a compelling solution due to its ability to address the complexities and challenges inherent in optimizing photovoltaic (PV) panel performance. MPPT is pivotal for maximizing energy production in solar systems by continuously adjusting operating point to the point of maximum power output, adapting to varying environmental conditions like irradiance levels, temperature fluctuations.

Random Forest Regression stands out as an apt choice for MPPT due to its capacity to handle nonlinear and complex relationships prevalent in solar energy data. Unlike traditional linear regression methods, RFR can capture intricate patterns and nonlinearities present in the relation between input temperature, irradiance, voltage & current variables and maximum power point, ensuring accurate predictions under diverse operating conditions.

Moreover, RFR exhibits robustness to overfitting, a common challenge in machine learning models, which ensures that the MPPT algorithm generalizes well to unseen data. This reliability is crucial for real-world applications where environmental conditions may vary widely this is further explained in the subsequent chapters of the thesis.

The ensemble nature of RFR, which combines multiple decision trees, further enhances its suitability for MPPT. By aggregating predictions from multiple trees, Random Forest Regression improves prediction accuracy and reduces variance, ensuring robust and reliable MPPT performance even in the presence of noise or uncertainty in the data.

Hence, Random Forest Regression emerges as a compelling choice for MPPT in solar energy systems, offering the capability to handle nonlinearity, robustness to overfitting, ensemble learning capabilities, and insights into feature importance. By leveraging these

strengths, RFR contributes to more accurate, reliable, and efficient MPPT algorithms, ultimately enhancing the viability and sustainability of solar energy systems.

3.6 Simulation & Data Capturing Software

3.6.1 MATLAB

MATLAB is a powerful tool commonly used in engineering and scientific fields for simulating and analyzing various systems. Maximum Power Point Tracking (MPPT), which plays a crucial role in improving the efficiency of solar systems by continuously regulating the operating point of solar panels to extract the maximum available power from the solar irradiance.

MATLAB provides a comprehensive environment for simulating PV systems. Using tools like Simulink, it can model the behavior of PV panels including irradiance, temperature and electrical characteristics (PV panel voltage and current). Simulink's library includes blocks specifically designed for PV system simulation, making it easier to construct complex models.

MATLAB allows to develop and test various MPPT algorithms. These algorithms range from conventional methods like Perturb and Observe (P&O) to more advanced techniques. MATLAB's extensive mathematical libraries and optimization tools facilitate the implementation and testing of these algorithms.

Most importantly MATLAB supports interfacing with hardware for real-time data acquisition, making it possible to capture data from sensors embedded in PV systems. Using tools like Simulink, it can interface with microcontrollers or data acquisition devices to capture real-time measurements such as voltage, current, irradiance & temperature from the PV panels.

3.6.2 Simulink

Simulink, is an integral part of MATLAB which offers a sophisticated visual platform tailored for modeling, simulating, and scrutinizing dynamic systems. With its intuitive interface it facilitates the creation of intricate models, rendering it an optimal solution for simulating Maximum Power Point Tracking (MPPT) algorithms.

Simulink offers a variety of blocks specifically designed for modeling PV systems. These blocks allow users to represent the electrical characteristics of PV panels, including the relationship between voltage, current, irradiance and temperature. By connecting these blocks together, users can create detailed models of entire PV arrays, taking into account factors such as shading and temperature effects.

Simulink allows to implement and test different MPPT algorithms within their PV system models. Whether it's conventional MPPT methods like Perturb and Observe (P&O) or more advanced techniques like Model Predictive Control (MPC). Simulink provides a flexible environment for designing and validating these algorithms. Further, users can easily compare the performance of different algorithms by swapping them in and out of their models.

Further, Simulink supports real-time simulation, enabling it to capture data from their PV system models as if they were operating in real-world conditions. This feature is particularly useful for testing MPPT algorithms under varying solar irradiance & temperature conditions. Users can monitor key parameters such as voltage, current & power output at real-time domain and analyze how their chosen MPPT algorithm responds to changes in environmental conditions.

Moreover, Simulink's visualization tools enable to analyze the performance of their MPPT algorithms through plots, scopes, and other graphical displays. Users can visualize the tracking behavior of different algorithms, compare the predicted and actual performance of their PV systems, and identify areas for improvement. Simulink also

integrates with MATLAB, allowing users to perform advanced data analysis and post-processing.

Most importantly Simulink facilitates HIL (Hardware-in-the-Loop) testing, where the PV system model is connected to physical hardware such as PV panels, inverters, and sensors. This allows users to validate their MPPT algorithms using real hardware under controlled laboratory conditions. Simulink provides seamless integration with hardware platforms, making it easy to interface with data acquisition devices and capture real-time measurements for analysis.

3.6 Summary

This chapter presented the adoption of machine learning-based Maximum Power Point Tracking (MPPT) techniques, particularly focusing on a regression-based model. This model integrates voltage, current, irradiance, and temperature measurements for enhanced MPPT efficiency. Challenges in obtaining relevant datasets led to data capture using MATLAB and Simulink, exported as CSV files for training. Python was chosen as the programming language for its versatility and extensive machine learning support. The scikit-learn library in Python facilitated model development, offering a wide range of algorithms and evaluation metrics. Random Forest Regression is chosen for MPPT in PV energy systems due to ability of handling complexities & nonlinear relationships in PV panel performance. MPPT optimizes energy production by adjusting the operating point to maximize power output under varying environmental conditions. MATLAB, alongside Simulink, provided a comprehensive environment for simulating PV systems, implementing MPPT algorithms, and conducting Hardware-in-the-Loop (HIL) testing. These tools enabled real-time data capture, visualization, and validation, crucial for developing and assessing machine learning based MPPT.

CHAPTER 4

NOVEL APPROACH TO MAXIMUM POWER POINT TRACKING (VCIT-MPPT)

4.1 Introduction

In chapter 3 justified the suitability of regression-based model in machine learning as the technology adopted for developing Maximum Power Point Tracking (MPPT). In this chapter novel approach to existing conventional MPPT was introduced and named as VCIT-MPPT, an acronym for voltage, current, irradiance, and temperature based MPPT. This chapter presents a novel approach by covering our hypothesis, input, output, process, features, and users and provide a brief summary of the methodology.

4.2 Hypothesis

Hypothesis of this research is Random Forest Regression based machine learning model which can be implemented for the task of maximum power point tracking of solar panel power generation system to optimize output power of solar panel. It involves designing and developing a machine learning model for MPPT that utilizes instantaneous voltage of the solar panel, instantaneous current drawn by the solar panel, irradiance of the incident wave, and temperature of the operating environment to accurately track the maximum power point of PV systems. Regression-based machine learning model will be designed to learn the relationships between the input parameters and the output and optimal operating point of the solar panel by replacing traditional MPPT control modules.

4.3 Input Process Output Model

In the Input Process Output (IPO) model utilizing Random Forest Regression where voltage (Vol), current (Cur), irradiance (Irr) and temperature (Temp) inputs are fed into the model. These inputs represent the key parameters affecting the performance of the

MPPT system. The Random Forest Regression algorithm processes these inputs and generates predictions for the reference voltage (V_{Ref}) value, which is crucial for optimizing the power output of the photovoltaic system.

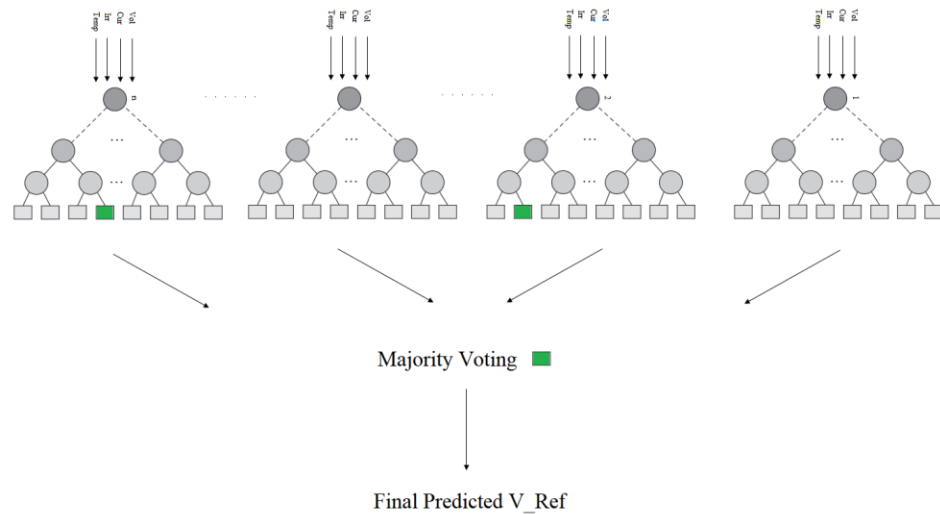


Figure 4.1: IPO model of VCIT-MPPT system

The Random Forest Regression algorithm operates by creating an ensemble of decision trees during training. Each decision tree is constructed based on a subset of the input data, and predictions are made by aggregating the results of multiple trees as shown in Figure 4.1. In the context of the VCIT-MPPT system, this ensemble approach helps in capturing the complex relationships between the input parameters (voltage, current, irradiance and temperature) and the desired output (reference voltage).

Once the RFR model is trained, it is capable of predicting the reference voltage value based on new input data. During the prediction process, each decision tree in the ensemble provides its own prediction for the reference voltage. The final prediction is determined by taking the majority vote among all the individual tree predictions.

This majority voting mechanism ensures that the final predicted reference voltage is robust and less susceptible to individual tree biases or outliers in the input data. By

aggregating predictions from multiple trees, the Random Forest Regression model reduces the risk of overfitting and enhances the generalization ability of the VCIT-MPPT system.

4.4 Input

Input to the machine learning model will be voltage of the solar panel measured via voltage sensor, current drawn by the solar panel measured via current transformer, irradiance of the incident waves via irradiance sensor, and temperature of the solar panel (ambient temperature) via temperature sensor. These parameters need to be obtained from said independent sensors in a PV system under various environmental conditions. The machine learning model will be trained on a dataset consisting of voltage, current, irradiance & temperature measurements obtained from respective sensors of a PV system under various environmental conditions and then the model will be validated. Once the model is validated it will be integrated with the PV panel and same said inputs of the PV panel (shown in Figure 4.1) which are voltage of the PV panel is measured via voltage sensor, current drawn by the solar panel measured via current transformer, irradiance of the incident waves via irradiance sensor, and temperature of the solar panel via temperature sensor will be provided to VCIT-MPPT system.

4.5 Output

Output of the model will be the reference voltage (V_{Ref}) signal is given to the Pulse Width Modulator (PWM) which needs to be compared with the triangular waveform from which the duty cycle or the width of the pulse (shown in Figure 4.1) will be decided and resulting adjusting the impedance of the solar panel to deliver maximum wattage of the solar panel.

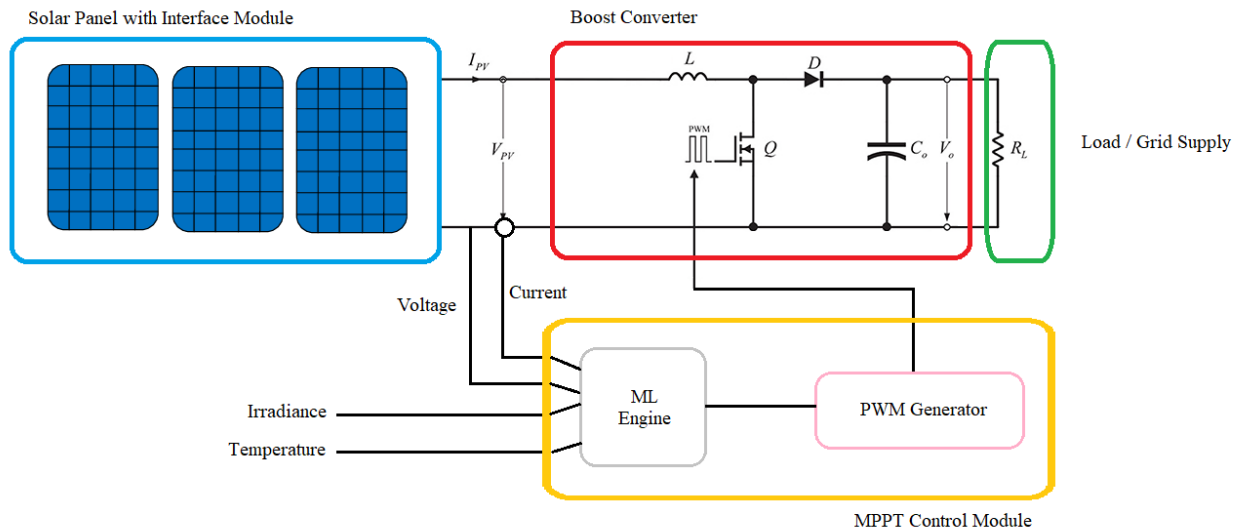


Figure 4.2: Inputs & Outputs of VCIT-MPPT system

4.6 Process

The overall process will include measuring instantaneous voltage of the solar panel via voltage sensor & measuring of instantaneous current drawn from the solar panel via a current transformer, measuring irradiance of the incident wave via an irradiance sensor, and temperature of the solar panel (ambient temperature) via temperature sensor. Then inputs the irradiance sensor and temperature sensor, PV panel voltage and the current transformer will be fed in to the model. During data acquisition process reference voltage (V_{Ref}) signal is given to the Pulse Width Modulator obtained from traditional MPPT control module should be fed in to the model to establish the relationship between input parameters and out parameters of the MPPT and to train the model. Required data can be obtained from the Simulink simulation environment in MATLAB. Once training is completed trained model needs to be validated using the same dataset. Then the traditional MPPT control module can be removed from the system and inputs from voltage of the solar panel measured via voltage sensor, current drawn by the solar panel measured via current transformer, irradiance of the incident waves via irradiance sensor, and temperature of the solar panel (ambient temperature) via temperature sensor should be given to the model. According to the input parameters, output of the model which is the reference voltage signal to the Pulse Width Modulator (PWM) which needs

to be compared with the triangular waveform from which the duty cycle or the width of the pulse will be generated and resulting adjusting the impedance of the PV panel to deliver maximum wattage from PV panel.

4.7 Features

This machine learning model is capable of recognizing and identifying patterns in datasets. By analyzing the data, models can discover complex relationships between voltage of the solar panel, instantaneous current drawn by the solar panel, irradiance of the incident wave, and environmental temperature. Further, its ability to adapt to the environmental changes which is useful in dynamic environments where irradiance and temperature patterns may change over time. Machine learning model can provide real-time power tracking by changing reference voltage value (V_{Ref}), allowing the PV system to quickly respond to changing of voltage, current, irradiance, and temperature parameters and optimize power generation. This enables efficient and effective operation of the solar panel.

4.8 Users

Users of the research outcome will be any solar power plant owners ranging from larger scale solar power plant industries to small scale domestic solar panel power plant consumers.

4.9 Summary

This chapter presented a novel approach to maximum power point tracking using RFR based machine learning model in solar panels. The hypothesis focuses on optimizing solar panel output using instantaneous voltage, current, irradiance, and temperature data. Inputs come from various sensors, and the model's output is a reference voltage signal for Pulse Width Modulation, adjusting solar panel impedance for maximum wattage. The process involves data acquisition, Simulink simulations for training, and validation.

Features include pattern recognition, adaptability to changing conditions, and real-time performance. Potential users are solar power plant owners, from large-scale industries to small-scale domestic consumers. The chapter 5 describes the design of the proposed machine learning based maximum power point tracking system.

CHAPTER 5

DESIGNING OF VCIT-MPPT

5.1 Introduction

In chapter 4 it was discussed the approach to the machine learning based VCIT-MPPT. The process discussed in the previous chapter is more elaborated in this chapter. This chapter describes about the design of a machine learning-based VCIT-MPPT system for solar power systems. It covers the top-level architecture of the VCIT-MPPT system, a sophisticated approach to optimizing solar panel performance. This chapter explains how machine learning algorithms, fed with critical parameters like voltage, current, irradiance, and temperature, dynamically adjust power converter settings in real-time to extract maximum power from solar panels amidst changing environmental conditions. Beginning with an overview of the system's architecture, the chapter delineates the roles of key modules such as the solar panel interface module, MPPT control module, and ML engine. It discusses how these modules interface, gather data, preprocess it, and make informed decisions for optimal power point tracking. Furthermore, it explores the selection of ML models crucial for accurate MPPT, detailing dataset analysis, model training, and performance evaluation using techniques like Random Forest Regression. This comprehensive exploration sets the stage for understanding the intricate workings and efficacy of the VCIT-MPPT system in maximizing solar energy conversion efficiency.

5.2 Top Level Architecture of VCIT-MPPT System

VCIT-MPPT, an acronym for voltage, current, irradiance & temperature based Maximum Power Point Tracking. Voltage, current, irradiance, and temperature-based machine learning MPPT (Maximum Power Point Tracking) system leverage these parameters as inputs to predict and track the optimal operating point for solar panels. By utilizing machine learning techniques, these algorithms can adaptively adjust the power converter's parameters in real-time to maximize the power output from the solar panels

under varying environmental conditions. The voltage, current, irradiance, and temperature are essential parameters that affect the power generation characteristics of solar panels. Input to the machine learning model will be voltage of the solar panel measured via voltage sensor, current drawn by the solar panel measured via current transformer, irradiance of the incident waves via irradiance sensor, and temperature of the solar panel (ambient temperature) via temperature sensor. Output of the model will be the reference voltage (V_{Ref}) signal given to the PWM which needs to be compared with the triangular waveform from which the duty cycle or the width of the pulse will be decided and resulting adjusting the impedance of the solar panel to deliver maximum wattage of the solar panel.

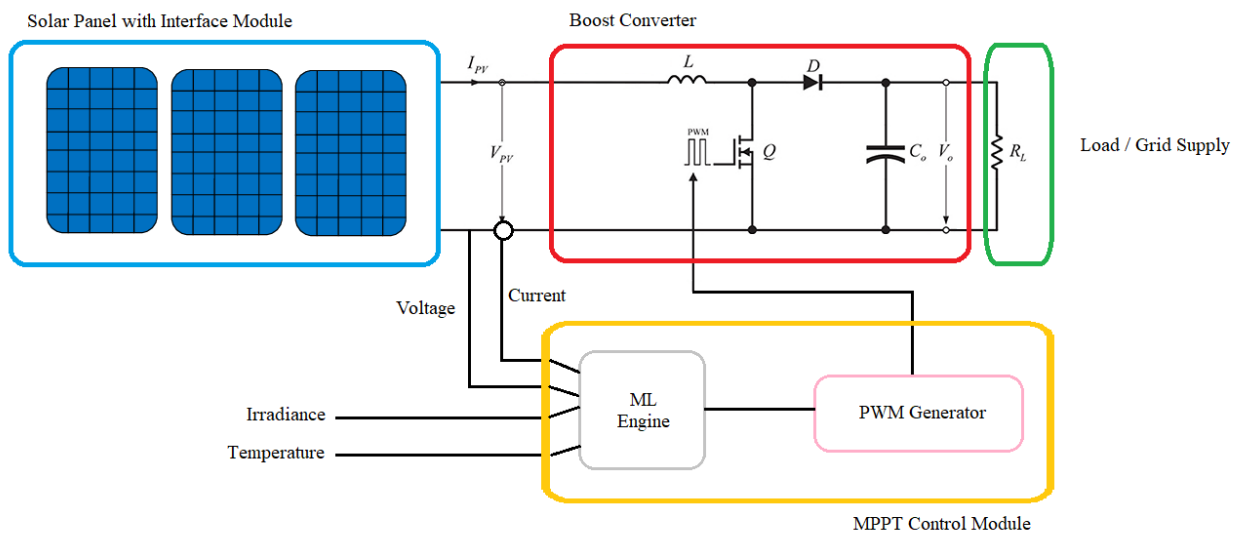


Figure 5.1: Top Level Architecture of VCIT-MPPT system.

Machine learning algorithms can analyze the relationship between voltage and power output to predict the optimal voltage for maximum power extraction. Current is the flow of electric charge through the solar panel. Similar to voltage, the current is influenced by environmental factors such as irradiance and temperature. Machine learning algorithms consider the relationship between current and power output to determine the optimal current for maximum power extraction. Irradiance refers to the quantity of solar

radiation incident on the solar panel's surface. It plays an important role in determining the power output of the panel.

Machine learning MPPT algorithms use historical data of voltage, current, and irradiance to learn the patterns and correlations, enabling them to predict the optimal power point based on real-time irradiance measurements. Temperature affects the efficiency and performance of solar panels. As temperature increases, the panel's output voltage and current may change, potentially reducing the power generation capability. ML algorithm consider the impact of temperature on voltage, current & power output to adjust MPPT parameters accordingly.

5.3 Modules in the Design

5.3.1 Solar Panel Interface Module

The Solar Panel Interface Module is a crucial component of the VCIT-MPPT system. Its primary function is to interface with the solar panels, gather the necessary data, and provide it to the ML engine for further processing. This module acts as a bridge between the physical solar panels and the ML algorithms. In MATLAB Simulink solar panel interface module is integrated to the PV panel is shown in Figure 5.2 and from the 'm' (measurement) port we can obtain the parameters. Typically, Solar Panel Interface Module consists of hardware and software components. On the hardware side, it typically includes sensors, analog-to-digital converters (ADCs), microcontrollers, and communication interfaces. The sensors are used to measure the solar panel's parameters, such as voltage, current, and temperature. The ADCs convert the analog signals from the sensors into digital data for processing. The microcontroller manages the data acquisition process and facilitates communication between the sensors, ADCs, and ML engine. The communication interfaces enable data transmission between the module and the ML engine.

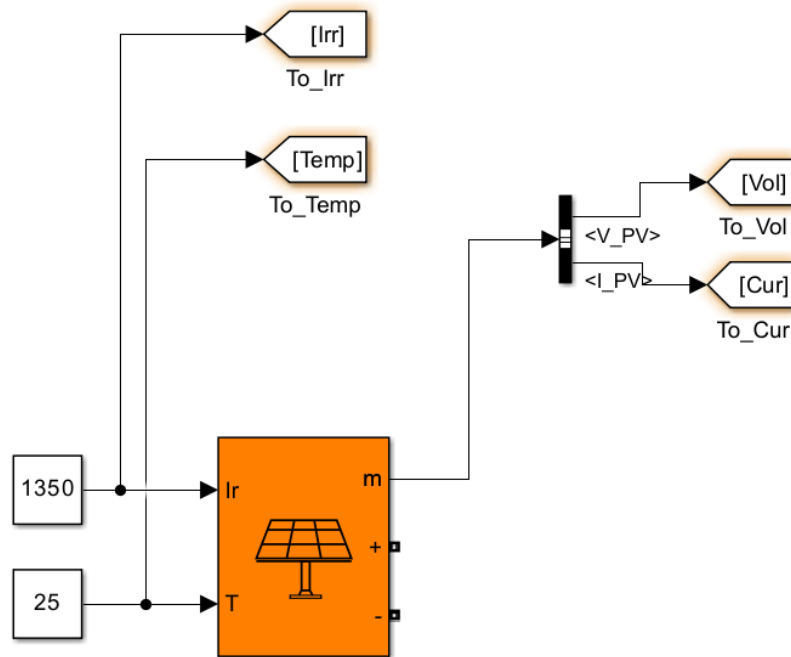


Figure 5.2: MALTAB Simulink solar panel interface module integrated to the PV panel

The module acquires data from the solar panels, including voltage, current, and temperature. These parameters are crucial for accurate MPPT operation and training the ML algorithms. The acquired data may be noisy or contain outliers, which can affect the performance of the ML algorithms. Therefore, preprocessing techniques should be applied to ensure reliable and high-quality data for subsequent stages.

By effectively interfacing with the solar panels and acquiring the necessary data, the Solar Panel Interface Module enables the ML engine to make informed decisions for achieving maximum power point tracking. The acquired data serves as the input for the ML algorithms, which then analyze and predict the optimum operating point of the solar panels based on the received information.

5.3.2 MPPT Control Module

The MPPT Control Module is an essential module of the VCIT-MPPT. It consists of ML Engine and Pulse Width Modulation (PWM) generator modules. Primary function of the

MPPT control module is to control and regulate the operation of the PV power system to ensure maximum power extraction from solar panels. This module integrates the outputs from the ML engine and generates control signals to adjust the operating conditions of the system. The MPPT control module incorporates the MPPT algorithm determined by the ML engine. The algorithm takes into account the real-time data received from the solar panel interface module and predicts the optimum operating point for maximum power extraction. The MPPT algorithm utilizes the characteristics of the solar panels, such as the voltage-current (V-I) curve, to track the maximum power point. However, in this research once ML engine is trained then, there will be no requirement of seeking the voltage-current (V-I) curve for MPPT since, same will be directly given by the ML engine. The MPPT algorithm determines the appropriate reference voltage that needs to be provided to the power converter to obtain maximum power extraction by varying the duty cycle. The control module interfaces with the power converter and adjusts the output voltage or current based on the algorithm's instructions.

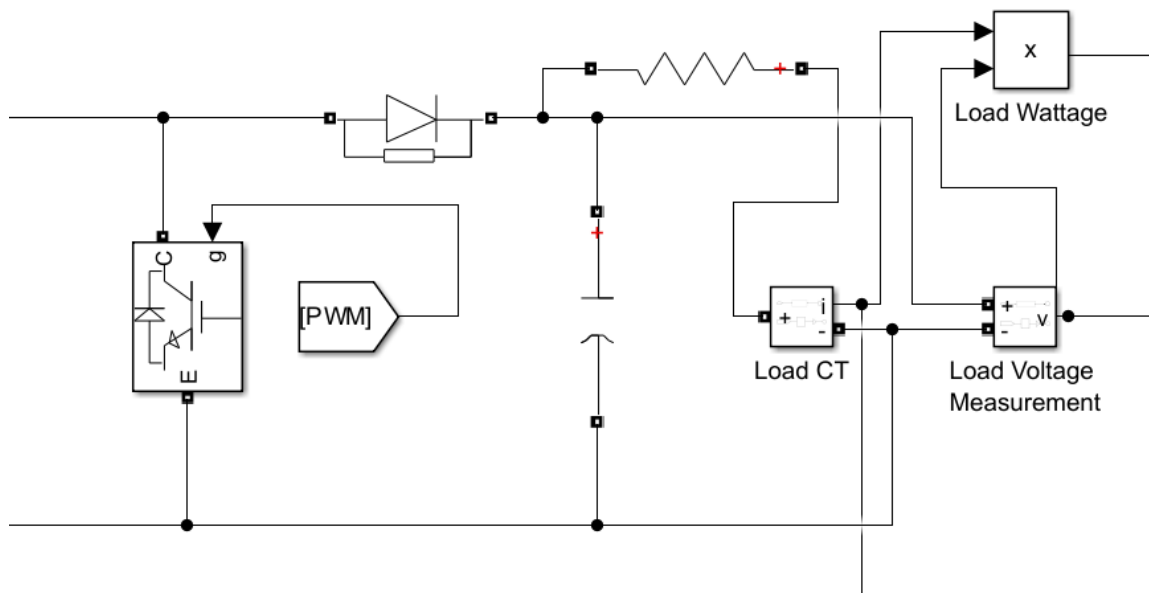


Figure 5.3: MPPT control module

The MPPT control module continuously monitors voltage of the solar panel via voltage sensor and measuring of the instantaneous current drawn from the solar panel via a current transformer, measuring irradiance of the incident wave via an irradiance sensor,

and temperature of the solar panel (ambient temperature) via temperature sensor. According to the input parameters, output of the model which is the reference voltage (V_{Ref}) signal to the Pulse Width Modulator (PWM) which needs to be compared with the triangular waveform from which the duty cycle or the width of the pulse will be generated and resulting adjusting the impedance of the solar panel. This ensures that the system operates at the maximum power point, even in the presence of variations in solar irradiance, temperature, or panel aging.

In addition to power tracking, the MPPT control module is accountable for regulating the output voltage & current of the solar power system. It ensures that the output voltage remains within the acceptable range defined by the system specifications. The module also monitors and regulates the output current to prevent overloading or underutilization of the system. By regulating the output voltage and current, the control module ensures the stability and safety of the PV system, optimizing its performance and preventing damage to connected loads / grid or the power converter.

Moreover, the MPPT control module acts as the decision-maker and controller in the VCIT-MPPT system. It receives input from the ML engine and adjusts the power converter's operating conditions to maximize power extraction from the solar panels. The module's real-time power tracking and voltage regulation capabilities without affecting the energy harvesting process enable efficient and reliable operation of the PV system, enhancing its overall energy conversion efficiency.

5.3.3 ML Engine

The ML Engine serves as the core component of the VCIT-MPPT system. This approach uses regression-based model. It encompasses model training and real-time inference to enable accurate and efficient maximum power point tracking. ML Engine is responsible for training the ML algorithms using historical data and making predictions for optimal power point tracking in real time. Historical data from the PV panel interface module,

containing information about solar irradiance, temperature, voltage, current are used for training purposes.

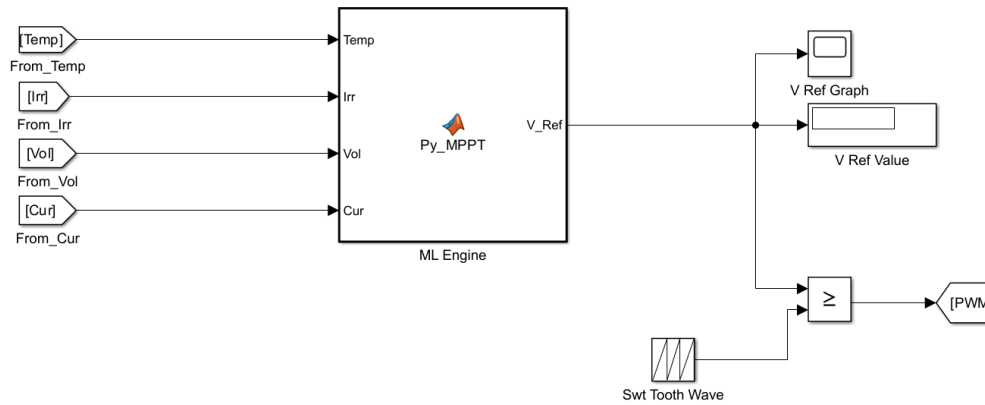


Figure 5.4: ML Engine designed in MATLAB Simulink

The ML engine conducts the training process, where training of the ML model is carried out from the preprocessed training data. During training, the model learns the patterns and relationships in the data to make accurate predictions of the optimal power point. Once the model is trained and validated, the ML engine enables real-time inference for maximum power point tracking. The module receives real time data from the PV panel interface module, including solar irradiance, temperature, voltage, and current measurements. Based on this data, the trained models make predictions of the optimal power point by adjusting reference voltage (V_{Ref}), guiding the MPPT control module in adjusting the power converter's operating conditions. Moreover, ML engine's real-time inference capabilities allow the MPPT system to adapt and respond to varying environmental conditions, ensuring accurate power tracking and maximizing the energy conversion efficiency of the solar power system.

5.4 Selection of ML Model

The selection of a machine learning (ML) model for the VCIT-MPPT system involves a meticulous process encompassing dataset analysis, visualization, and model evaluation. Initially, the dataset underwent preprocessing, including handling missing values

through forward fill (ffill) and ensuring completeness using DataFrame.describe(). Subsequently, data normalization and visualization were performed using seaborn's pairplot() function to explore relationships between variables. Correlation analysis, visualized with seaborn's heatmap(), unveiled significant associations among inputs like irradiance, temperature, voltage, current, and the output, reference voltage. The analysis revealed strong positive correlations between voltage reference and current (0.95) and irradiance (0.92), indicating simultaneous increases. Conversely, weaker relationships were observed between voltage reference and temperature (0.09) and voltage (-0.28), suggesting less pronounced or inverse associations.

Several regression models, including Linear Regression, Ridge Regression, Lasso Regression, Decision Tree Regression, and Random Forest Regression, were trained, evaluated, and compared using metrics like Mean Squared Error (MSE) and R-squared (R²). Random Forest Regression emerged as the optimal choice due to its superior performance, characterized by lower MSE and higher R² values. This decision aligns with the method's robustness, capacity to handle complex relationships, and resistance to overfitting. By selecting Random Forest Regression, the VCIT-MPPT system stands to benefit from improved accuracy and efficiency in maximum power point tracking. The ensemble learning approach of Random Forests, which aggregates predictions from multiple decision trees, mitigates overfitting concerns and enhances predictive accuracy. This comprehensive approach ensures that the chosen model is well-suited for the specific requirements of the VCIT-MPPT system, leading to optimized performance and functionality.

5.5 Model Description

Designed model is a machine learning model designed for predicting a variable named voltage reference (V_Ref) based on the given input features voltage (Vol), current (Cur), irradiance (Irr) and temperature (Temp). This model requires libraries like NumPy, Pandas, and scikit-learn modules for data manipulation, visualization, and machine learning algorithms. First model needs to loads the dataset in comma separated values

(MPPT_Dataset.csv) into a Pandas DataFrame named 'historical_data' and performs some basic data preprocessing steps like dropping unnecessary columns and handling missing values by forward filling. Then, it is required to conduct exploratory data analysis using Seaborn to visualize the relationships between variables with pair plots and correlation heatmap. Once analysis part is completed the dataset needs to be splitted into input features (X) and target variable (y) and further divides it into training and testing sets using the train_test_split function from scikit-learn.

Then it is required to initialize RandomForestRegressor model, and need to fits it to the training data, and makes predictions on the test data. After predicting the values, it creates a DataFrame to compare the actual and predicted values and visualized using a bar plot. Subsequently, the model will calculate variance and bias and visualizes the relationship between actual and predicted values using a scatter plot with a diagonal line representing perfect predictions. Once model is trained by defining MPPT function which takes input parameters features voltage (Vol), current (Cur), irradiance (Irr) and temperature (Temp) will predict the reference voltage (V_Ref) value using the trained ML model.

5.6 Summary

The VCIT-MPPT system employs a top-level architecture integrating machine learning for MPPT in solar panels. This system utilizes voltage, current, irradiance, and temperature data to optimize power extraction. The ML algorithms analyze historical data to predict optimal operating points in real time. The architecture comprises three main modules: the solar panel interface module, MPPT control module, and ML engine. The solar panel interface module interfaces with solar panels, gathering crucial data like voltage, current, and temperature. This data undergoes preprocessing before being fed into the ML engine. The MPPT control module regulates the power converter based on ML algorithm outputs, ensuring maximum power extraction while maintaining system stability. It continuously monitors panel parameters and adjusts the system accordingly. The ML engine is the core component, responsible for training and real time inference. It

utilizes historical data to train ML models and makes predictions for optimal power points, guiding the MPPT Control Module. The ML model selection process involves dataset analysis, visualization, and evaluation of various regression models. Random Forest Regression emerged as the optimal choice due to its robustness and superior performance in predicting the reference voltage. The designed model utilizes libraries like NumPy, Pandas, and scikit-learn for data manipulation and machine learning tasks. It undergoes training and testing phases, with performance evaluation and visualization of results. Ultimately, the trained ML model enables accurate and efficient MPPT in the VCIT-MPPT system, enhancing overall energy conversion efficiency. Chapter 6 describes the implementation process of the proposed machine learning based VCIT-MPPT system.

CHAPTER 6

IMPLEMENTATION

6.1 Introduction

In chapter 5 it was discussed the designing of machine learning-based VCIT-MPPT. This chapter describes about the implementation of a machine learning-based VCIT-MPPT system for solar power systems which covers the VCIT MPPT System. Further, technical setup and software installation process for implementing a machine learning-based VCIT-MPPT system are detailed. The system utilizes an Core i7 CPU with 32.0 GB RAM, operating on Windows 10 Pro (version 22H2). Essential software components include Python 3.11, Jupyter Notebook 6.4.12, MATLAB R2023b, and Simulink 23.2. The data acquisition process involves simulating conventional MPPT using MATLAB Simulink software, capturing parameters like instantaneous voltage, current, irradiance, and temperature. Dataset generation and preprocessing steps are outlined, including the conversion to CSV format for further analysis. Model training using Random Forest Regression is explained, emphasizing the selection of libraries and handling of missing values. The performance evaluation of the trained model showcases impressive accuracy and minimal error metrics. Additionally, the function for predicting voltage reference (V_{Ref}) using the MPPT algorithm is defined, along with its integration into MATLAB Simulink through the Python function block (Py_MPPT), enabling seamless execution of custom Python code within the simulation environment.

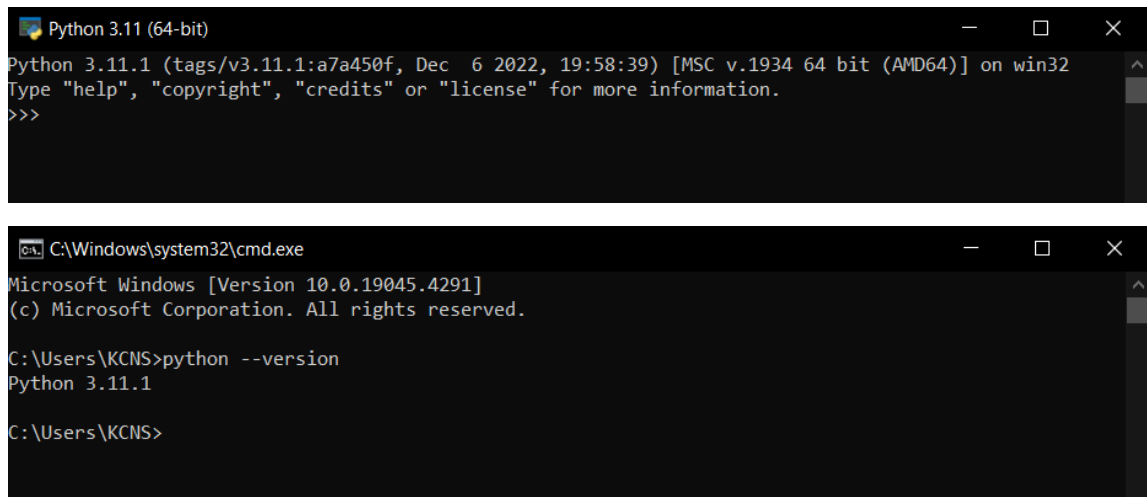
6.2 OS & Platforms

To implement machine learning-based VCIT-MPPT Intel(R) Core (TM) i7-10510U CPU @ 1.80GHz 2.30 GHz computing infrastructure with 32 Giga Bytes of Random Access Memory was used. Windows 10 Pro (22H2) was installed as the operating system which supports Python 3.11, Jupyter Notebook 6.4.12, MATLAB R2023b and Simulink 23.2 and same were installed on the computing infrastructure.

6.3 Installation of Required Software

6.3.1 Installation of Python 3.11

Download the installer for Python from the official Python website (<https://www.python.org>) and downloading the Python 3.11 (or latest if required) installer suitable for Windows. It's crucial to choose the correct version, ensuring compatibility with the system architecture (32-bit or 64-bit). Once the installer is downloaded, double-click on the executable file to initiate the installation process. The Python installer wizard will guide through the installation steps, allowing customize certain options, such as adding Python to the system PATH for easy command-line access. Proceed with the installation, and once completed, Python 3.11 will be successfully installed on the system. To verify the installation, open Command Prompt or PowerShell and enter the command "python --version" to confirm that Python 3.11 has been installed and is accessible from the command line.



```
Python 3.11 (64-bit)
Python 3.11.1 (tags/v3.11.1:a7a450f, Dec 6 2022, 19:58:39) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>>

C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.4291]
(c) Microsoft Corporation. All rights reserved.

C:\Users\KCNS>python --version
Python 3.11.1

C:\Users\KCNS>
```

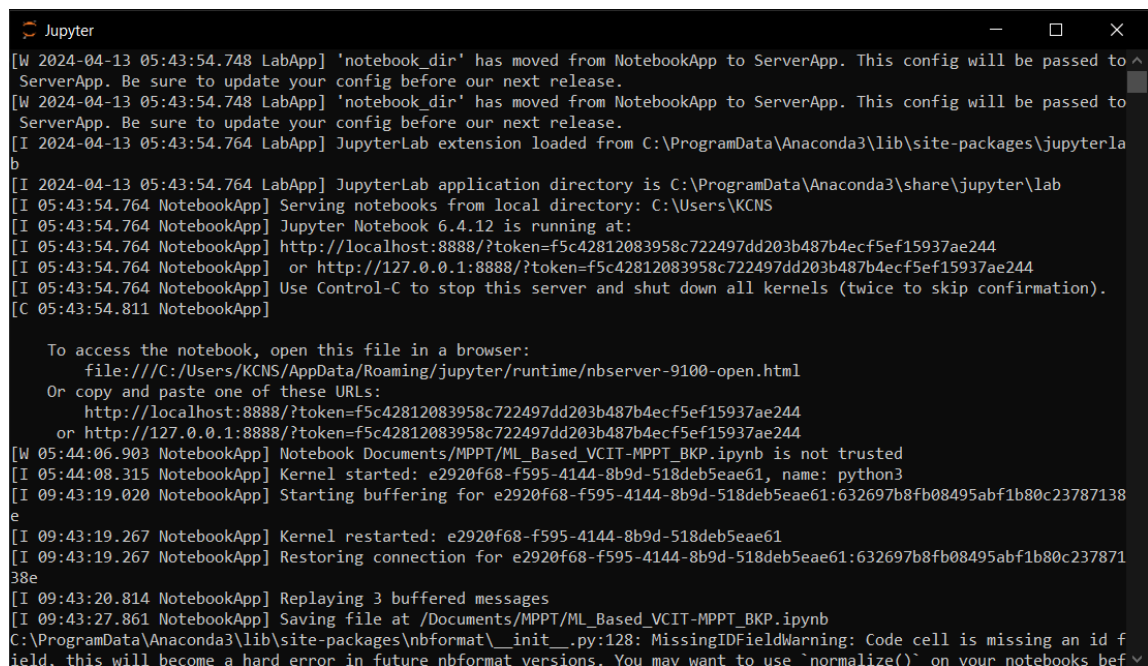
Figure 6.1: Checking the Python version installed.

Additionally, it is necessary to set up essential Python libraries essential for this study, including NumPy (for handling multi dimensional arrays and matrices, and performing high level mathematical operations), pandas (for data manipulation and analysis), math

(for basic mathematical functions), matplotlib (for generating plots), seaborn (for data visualization, built on top of matplotlib), and sklearn (a machine learning library providing support for classification, regression, and clustering algorithms). These libraries can be installed via the Preferred Installer Program, commonly known as PIP..

6.3.2 Installation of Jupyter Notebook 6.4.12

Installing Jupyter Notebook offers users a dynamic platform for interactive computing, enabling seamless integration of code, visualizations, and narrative text. Ensure Python is installed on your system, as Jupyter Notebook is primarily a Python package. Utilizing package managers such as pip (Python's package installer) simplifies the installation process. By executing the command "pip install jupyter notebook" in the command prompt or terminal, even it can swiftly install Jupyter Notebook along with its dependencies. Once installed, initiating Jupyter Notebook involves running the command "jupyter notebook" in the terminal, which starts the Jupyter server and launches the notebook interface in a web browser.



```
Jupyter
[W 2024-04-13 05:43:54.748 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to
ServerApp. Be sure to update your config before our next release.
[W 2024-04-13 05:43:54.748 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to
ServerApp. Be sure to update your config before our next release.
[I 2024-04-13 05:43:54.764 LabApp] JupyterLab extension loaded from C:\ProgramData\Anaconda3\lib\site-packages\jupyterlab
[I 2024-04-13 05:43:54.764 LabApp] JupyterLab application directory is C:\ProgramData\Anaconda3\share\jupyter\lab
[I 05:43:54.764 NotebookApp] Serving notebooks from local directory: C:\Users\KCNS
[I 05:43:54.764 NotebookApp] Jupyter Notebook 6.4.12 is running at:
[I 05:43:54.764 NotebookApp] http://localhost:8888/?token=f5c42812083958c722497dd203b487b4ecf5ef15937ae244
[I 05:43:54.764 NotebookApp] or http://127.0.0.1:8888/?token=f5c42812083958c722497dd203b487b4ecf5ef15937ae244
[I 05:43:54.764 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 05:43:54.811 NotebookApp]

To access the notebook, open this file in a browser:
file:///C:/Users/KCNS/AppData/Roaming/jupyter/runtime/nbserver-9100-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=f5c42812083958c722497dd203b487b4ecf5ef15937ae244
or http://127.0.0.1:8888/?token=f5c42812083958c722497dd203b487b4ecf5ef15937ae244
[W 05:44:06.903 NotebookApp] Notebook Documents/MPPT/ML_Based_VCIIT-MPPT_BKP.ipynb is not trusted
[I 05:44:08.315 NotebookApp] Kernel started: e2920f68-f595-4144-8b9d-518deb5eae61, name: python3
[I 09:43:19.020 NotebookApp] Starting buffering for e2920f68-f595-4144-8b9d-518deb5eae61:632697b8fb08495abf1b80c23787138e
[I 09:43:19.267 NotebookApp] Kernel restarted: e2920f68-f595-4144-8b9d-518deb5eae61
[I 09:43:19.267 NotebookApp] Restoring connection for e2920f68-f595-4144-8b9d-518deb5eae61:632697b8fb08495abf1b80c23787138e
[I 09:43:20.814 NotebookApp] Replaying 3 buffered messages
[I 09:43:27.861 NotebookApp] Saving file at /Documents/MPPT/ML_Based_VCIIT-MPPT_BKP.ipynb
C:\ProgramData\Anaconda3\lib\site-packages\nbformat\_init_.py:128: MissingIDFieldWarning: Code cell is missing an id f
ield, this will become a hard error in future nbformat versions. You may want to use 'normalize()' on your notebooks bef
```

Figure 6.2: Jupyter server.



Figure 6.3: Jupyter notebook interface in a web browser.

6.3.3 Installation of MATLAB R2023b and Simulink 23.2

Download the installers for MATLAB R2023b and Simulink 23.2 from the MathWorks website (<https://www.mathworks.com>), accessible upon logging in with your MathWorks account. Initiate the installation process by running the installer executable. The installation wizard will guide through various configuration options, allowing to customize the installation to suit specific needs. Options typically include selecting the installation directory, choosing components to install (such as MATLAB, Simulink, toolboxes, etc.), and creating shortcuts for convenient access.

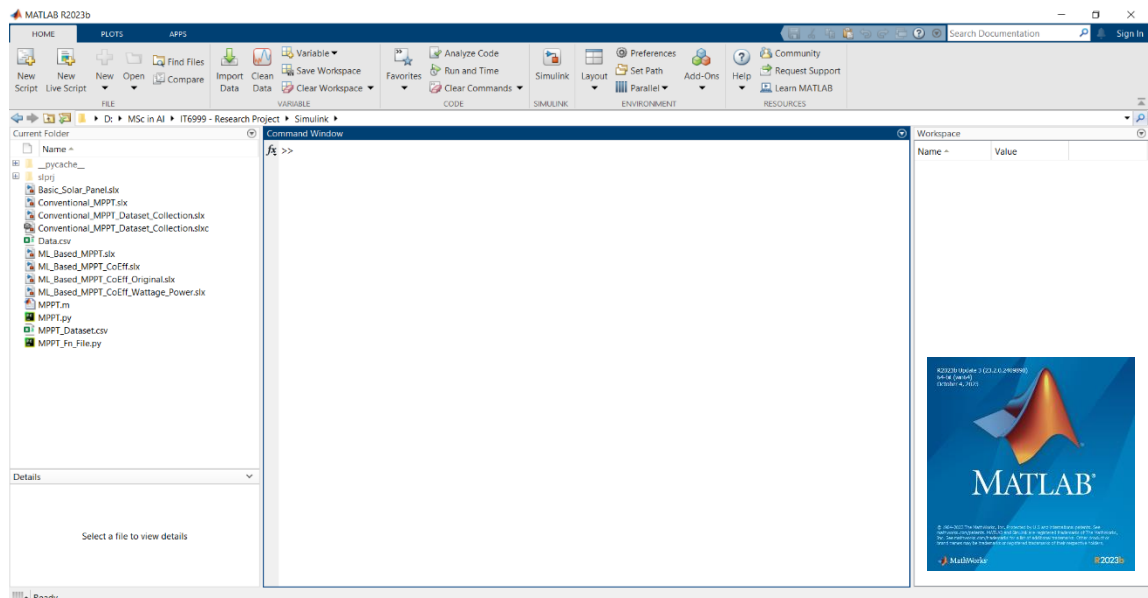


Figure 6.4: MATLAB R2023b.

Once installation is completed then it is required to activate MATLAB license or logging in with your MathWorks account credentials. Activation ensures compliance with licensing agreements and unlocks the full functionality of MATLAB R2023b and Simulink 23.2.

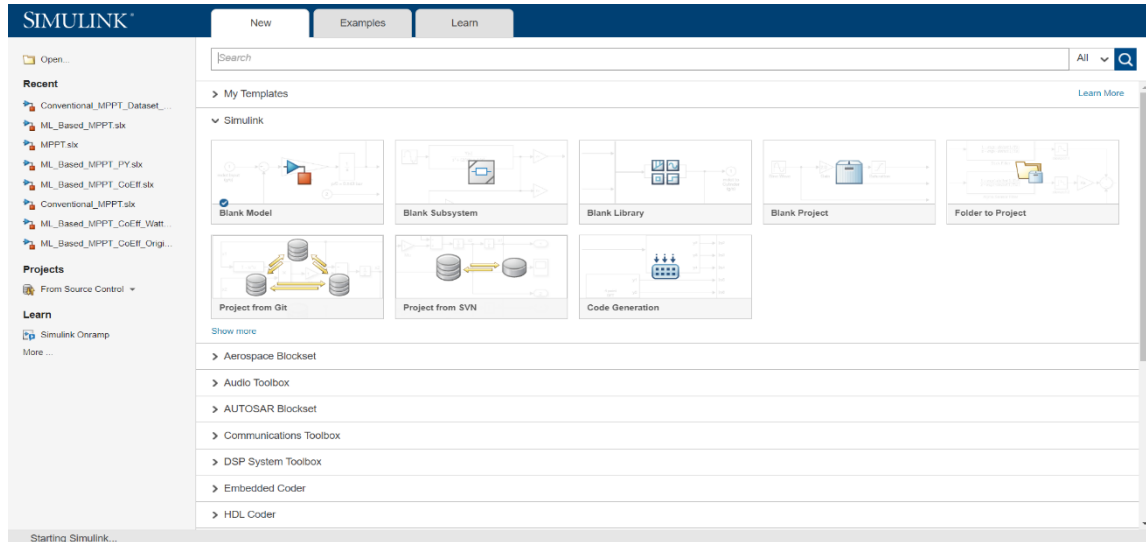


Figure 6.5: Simulink 23.2 installed on MATLAB R2023b.

6.4 Data Acquisition

Figure 6.6 shows the conventional Maximum Power Point Tracking (MPPT) which was simulated in the MATLAB Simulink software simulator which is required for capturing of required instantaneous voltage and instantaneous current drawn from the solar panel, irradiance of the incident wave and ambient temperature of the solar panel.

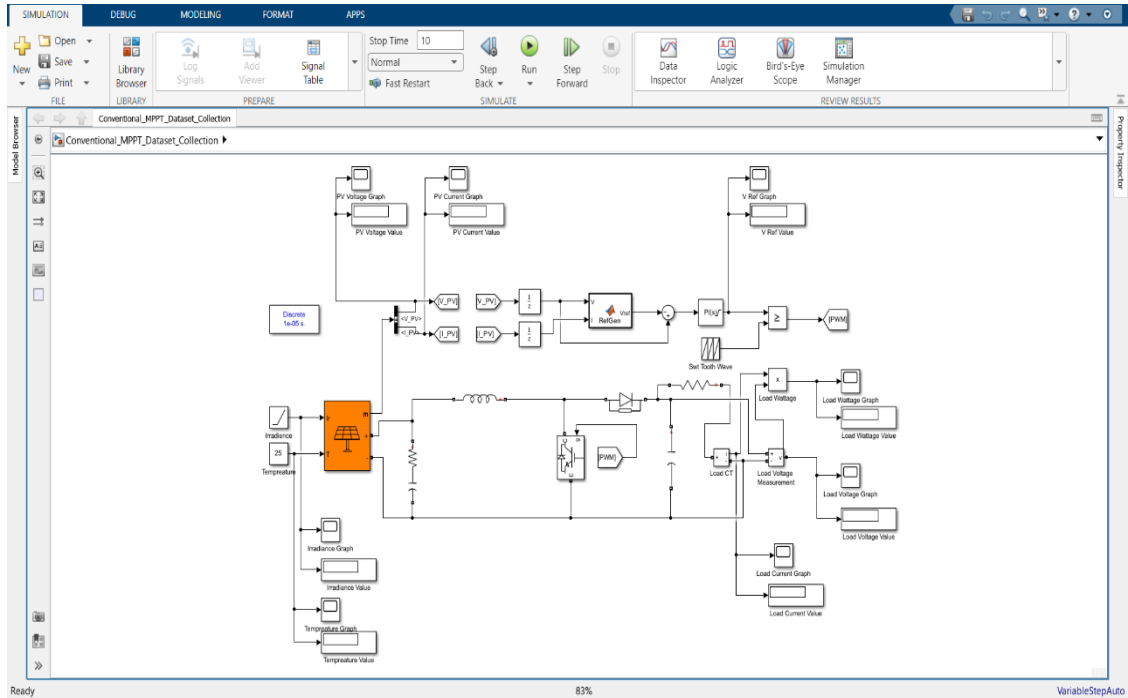


Figure 6.6: Conventional MPPT simulated in the MATLAB Simulink software simulator.

In order to convert signals in the output scope in the MATLAB Simulink, default settings of the has to change to data logging mode as shown in Figure 6.7.

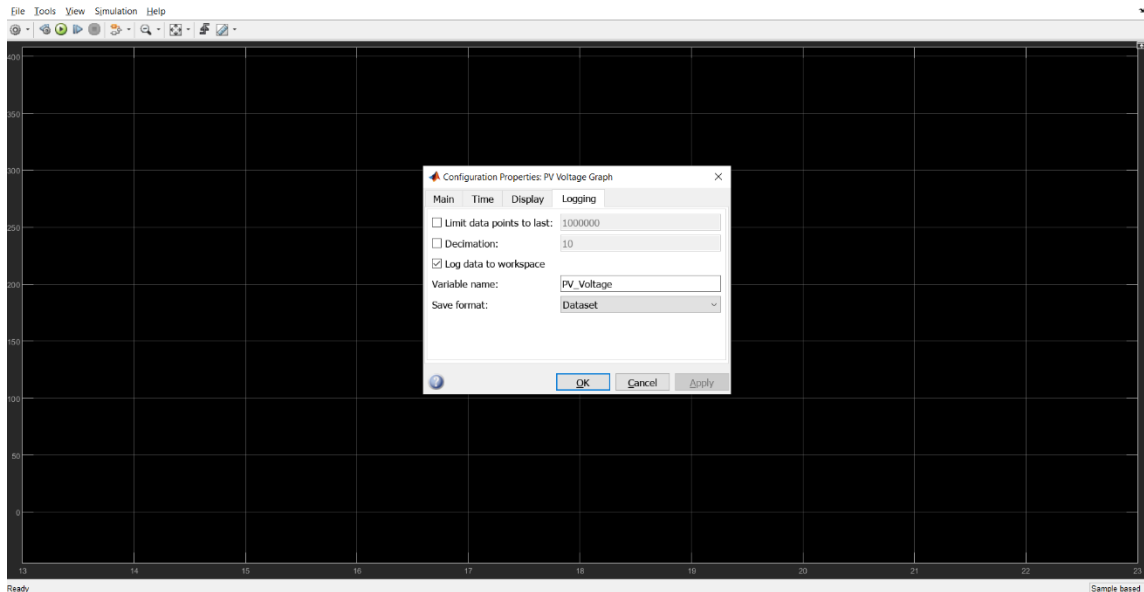


Figure 6.7: MATLAB Simulink software simulator data logging mode.

Data logging mode has to be enabled for the following parameters of solar panel. Instantaneous voltage of the solar panel, instantaneous current drawn from the solar panel, irradiance of the incident wave and ambient temperature of the solar panel.

Once settings are completed MATLAB Simulink can start the simulation process as shown in Figure 6.8.

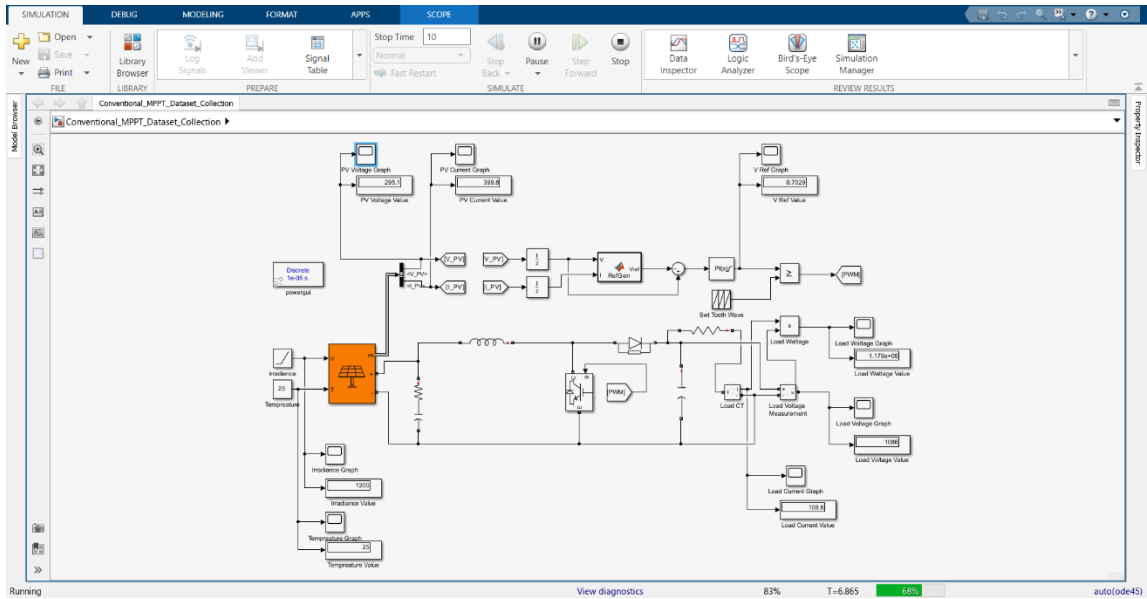


Figure 6.8: Conventional MPPT simulated in MATLAB Simulink software simulator with enabled data logging features.

6.3 Generating Dataset

Generating CSV file is an important step in using the MATLAB. Once the simulation is completed results can be found under the Workspace window of MATLAB as shown in Figure 6.9.

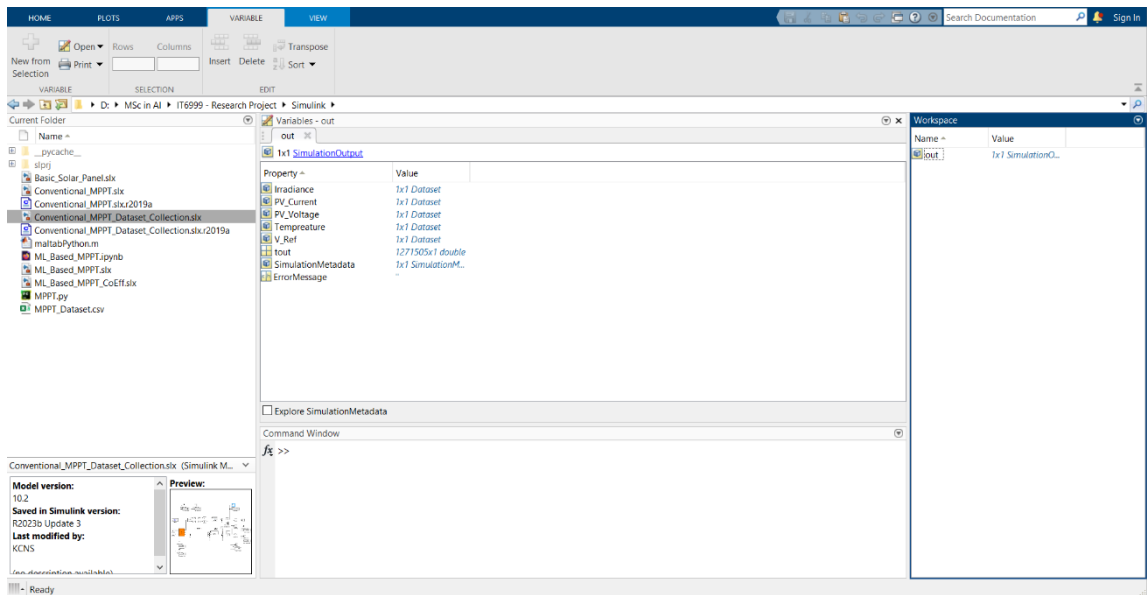


Figure 6.9: Output dataset displayed in MATLAB.

Once output data sets of instantaneous voltage of the solar panel, instantaneous current drawn from the solar panel, irradiance of the incident wave and ambient temperature of the solar panel were generated, they are required to compile to one dataset. MATLAB is capable of generating dataset table using table command as shown in the Figure 6.10.

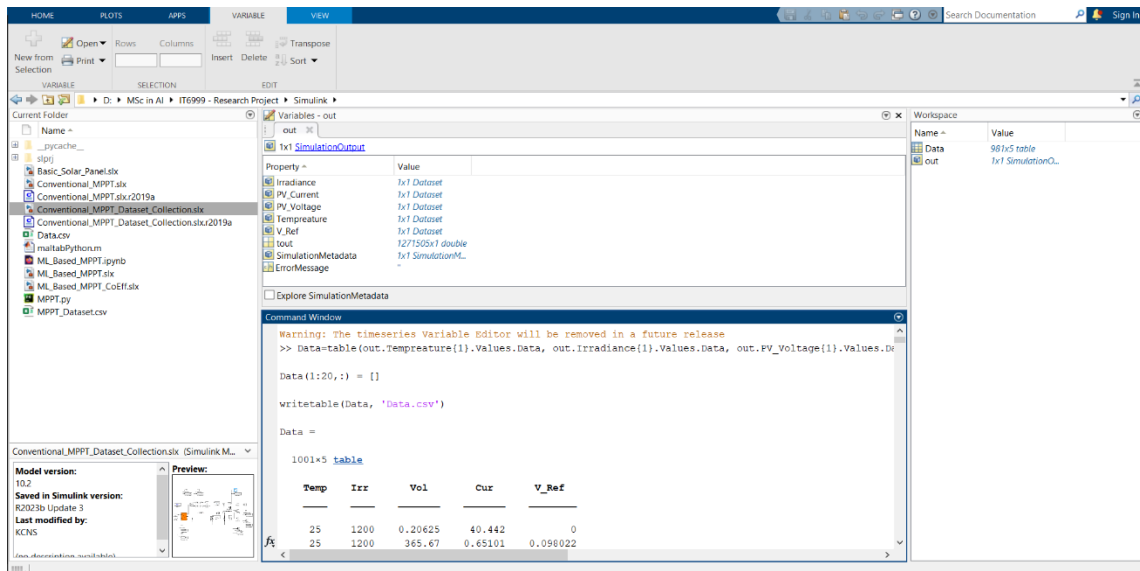


Figure 6.10: Generating dataset table in MATLAB.

CSV file can be created using a by writing a MATLAB script shown in Figure 6.11 to extract the necessary information from the MATLAB Workspace.

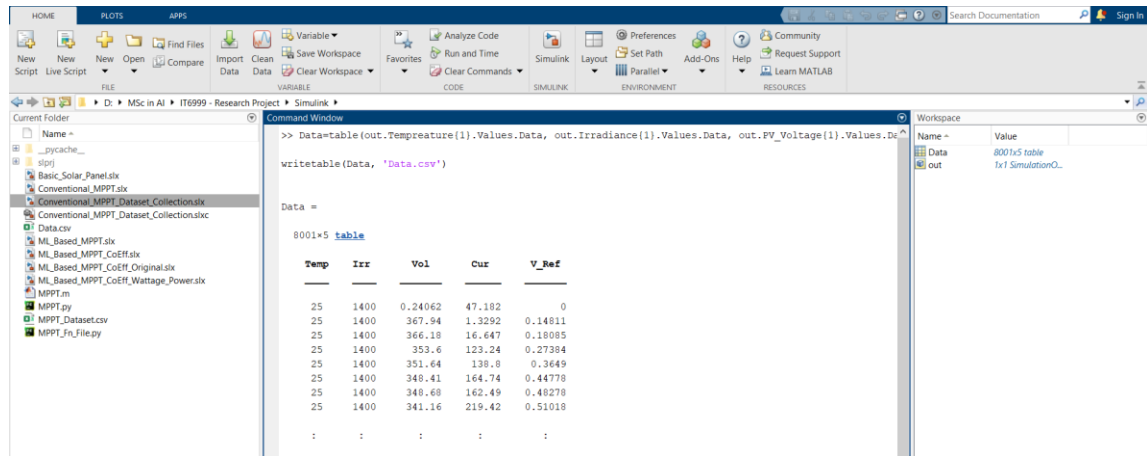


Figure 6.11: MATLAB script to generate CSV dataset.

Once the table is generated it can be converted to Comma Separated Values (CSV) file and exported as shown in Figure 6.12.

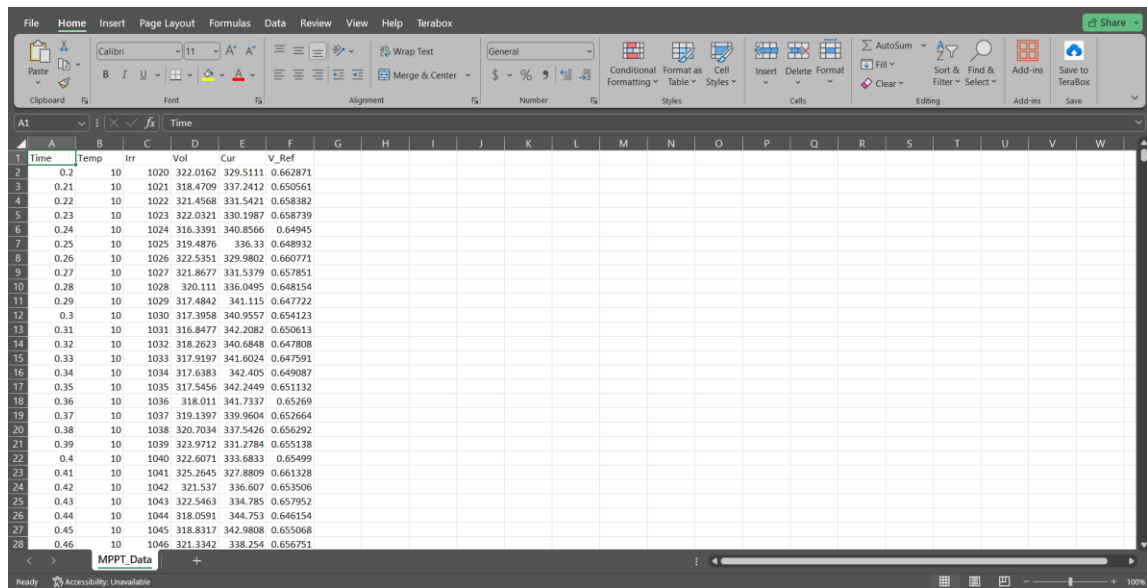


Figure 6.12: CSV dataset for conventional MPPT using MATLAB.

6.4 Model Training Process

Once dataset is obtained then step is to train the machine learning model. In chapter 5 it was explained that selection of reasons why Random Forest Regression was selected as the machine learning model for VCIT-MPPT system. First following Python libraries and machine learning libraries (numpy, pandas, math, pyplot, seaborn, sklearn) should be imported as indicated in the Figure 6.13.

```
In [2]: import numpy as np
import pandas as pd
import math
import warnings
import matplotlib.pyplot as plt
import seaborn as sb
from sklearn import metrics
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import accuracy_score
from sklearn.metrics import mean_squared_error
from sklearn.metrics import r2_score
warnings.filterwarnings("ignore")
```

Figure 6.13: Importing of Python libraries.

Then data set to be imported and converted it to Pandas data frame as shown in Figure 6.14.

```
In [3]: historical_data = pd.read_csv('MPPT_Dataset.csv')
In [4]: historical_data
Out[4]:
```

	Time	Temp	Irr	Vol	Cur	V_Ref
0	0.01	15	610.0	317.911931	199.255814	0.549306
1	0.02	15	610.1	317.284976	199.655029	0.555423
2	0.03	15	610.2	318.819654	197.625885	0.557012
3	0.04	15	610.3	320.937705	194.571333	0.559110
4	0.05	15	610.4	321.851733	193.398271	0.559641
...
165916	1659.17	35	1399.6	288.187668	439.726761	0.702669
165917	1659.18	35	1399.7	287.051466	443.632603	0.703589
165918	1659.19	35	1399.8	287.619457	442.633063	0.701525
165919	1659.20	35	1399.9	289.640819	434.713469	0.704073
165920	1659.21	35	1400.0	288.637305	438.262848	0.703160

165921 rows × 6 columns

Figure 6.14: Importing of dataset and converting to DataFrame.

Then dataset should be checked for missing values and if there are any missing values the "ffill" method is particularly useful in scenarios where missing data points occur

intermittently but can be reasonably assumed to have the same value as the preceding valid data point. When applying "ffill" to fill missing data, the last observed value is propagated forward to fill the gaps until the next valid data point is encountered. In MPPT datasets, using "ffill" ensures that missing data points do not disrupt the continuity of the time series. This approach maintains the integrity of the dataset and allows for the accurate analysis and optimization of the maximum power point (MPP) of a photovoltaic system over time, thereby improving the reliability and effectiveness of photovoltaic systems.

```
In [8]: historical_data.isnull().sum()
Out[8]: Temp    0
        Irr     0
        Vol    0
        Cur    0
        V_Ref  0
        dtype: int64

In [9]: data_set = historical_data.fillna(method='ffill')
```

Figure 6.15: Checking for missing values and applying "ffill" to fill missing data.

After preprocessing the dataset, it should be divided into separate training and testing sets in order to assess the performance of the model.

```
In [63]: X = data_set[['Temp', 'Irr', 'Vol', 'Cur']]
        y = data_set['V_Ref']

In [65]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)

In [66]: X_train
Out[66]:
```

	Temp	Irr	Vol	Cur
77930	24	1292.1	294.931232	430.043893
70038	23	1293.0	292.965168	436.820065
148921	33	1280.3	284.816520	424.572538
80717	25	780.7	311.867060	237.904505
143146	33	702.8	294.653736	228.047876
...
97639	27	892.7	300.880901	289.808823
96939	27	722.7	299.797076	238.940516
152315	34	829.6	278.468367	288.860647
117952	29	1343.8	294.762564	429.408789
43567	20	1016.2	306.992459	332.805966

```
116144 rows x 4 columns

In [67]: X_train.shape
Out[67]: (116144, 4)
```

Figure 6.16: Splitting the MPPT dataset into training and testing.

Then MPPT dataset has to be splitted into two parts X and y. X subset of the dataset consists of the input parameters of the temperature, irradiance, voltage, current values and the y subset of the dataset consists of output parameter of reference voltage.

```
In [79]: regressor = RandomForestRegressor()
In [80]: regressor.fit(X_train, y_train)
Out[80]: RandomForestRegressor()
In [81]: predicted = regressor.predict(X_test)
In [82]: predicted.shape
Out[82]: (49777,)
```

```
In [83]: dframe = pd.DataFrame(y_test, predicted)
In [84]: dfr = pd.DataFrame({'Actual V Ref': y_test, 'Predicted V Ref': predicted})
In [85]: dfr
Out[85]:
```

	Actual V Ref	Predicted V Ref
42423	0.657781	0.658984
97784	0.652105	0.656762
96434	0.610479	0.608037
99788	0.654063	0.654136
52417	0.655765	0.656089

Figure 6.17: Training the VCIT-MPPT model using Random Forest Regression.

Once VCIT-MPPT model is trained using Random Forest Regression it is required to test and evaluate its performance. For this variance and bias were calculated and model has given 0.0014319905179881916 value as variance and 1.2697465569407419e-05 value as bias. These figures indicate that the model exhibits minimal fluctuation and extremely slight prejudice. This implies that the model is sturdy (with low fluctuation) and adept at discerning the inherent patterns within the data (with low prejudice). Consequently, the model is expected to excel not only on the training dataset but also on unseen data, achieving a harmonious equilibrium between grasping intricate patterns and steering clear of overfitting.

```
In [94]: print('Accuracy: ',regressor.score(X_test, y_test))
Accuracy: 0.991115568752682
In [95]: print('Mean Abs Error: ',metrics.mean_absolute_error(y_test, predicted))
Mean Abs Error: 0.0024423303667789925
In [96]: print('Mean Sq Error: ',metrics.mean_squared_error(y_test, predicted))
Mean Sq Error: 1.2722421303877387e-05
In [97]: print('Root Mean Sq Error: ',math.sqrt(metrics.mean_squared_error(y_test, predicted)))
Root Mean Sq Error: 0.003566850333820777
```

Figure 6.18: Accuracy of the VCIT-MPPT model

The performance metrics provided for the VCIT-MPPT system utilizing Random Forest Regression are quite impressive. With an accuracy of 99.11 %, the system demonstrates a high level of precision in predicting the maximum power point (MPP) of a PV system based on voltage, current, irradiance, and temperature inputs. This indicates that the model's predictions closely match the actual MPP values, which is critical for improving the power output of PV panels.

Moreover, low values for Mean Absolute Error (MAE), Mean Square Error (MSE), and Root Mean Square Error (RMSE) validate the effectiveness of the model. MAE value 0.0024 suggests that, on average, the model's predictions deviate from the actual MPP by only a small margin. The MSE, being on the order of 10^{-5} , indicates minimal variance between predicted and actual values, while the RMSE of 0.0035 signifies the square root of this variance, emphasizing the model's overall accuracy.

Hence, these performances suggest that the VCIT-MPPT system utilizing Random Forest Regression is highly reliable and robust. It can effectively track the MPP of PV systems under variable environmental conditions, providing accurate predictions that can optimize energy harvesting and enhance the efficiency of solar power generation.

6.5 Predicting the Voltage Reference

MPPT_Fn is defined as serve as a Maximum Power Point Tracking (MPPT) function. This function takes four parameters: Temp (temperature), Irr (irradiance), Vol (voltage), and Cur (current). These parameters represent the real time environmental conditions & electrical characteristics of photovoltaic system, which are crucial for determining the maximum power point (MPP) of the system.

```
In [98]: def MPPT_Fn(Temp, Irr, Vol, Cur):  
        Inst_Temp = Temp  
        Inst_Irr = Irr  
        Inst_Vol = Vol  
        Inst_Cur = Cur  
  
        predicted_val = regressor.predict(np.array([[Inst_Temp, Inst_Irr, Inst_Vol, Inst_Cur]]))  
  
        return float(predicted_val)
```

Figure 6.19: Defining MPPT function in VCIT-MPPT.

Inside the MPPT_Fn function, the input parameters are assigned to corresponding variables: Inst_Temp, Inst_Irr, Inst_Vol, and Inst_Cur, respectively. These variables represent the instantaneous values of temperature, irradiance, voltage, and current, which are used as inputs for the MPPT algorithm. The function then utilizes the Random Forest Regressor to predict the V_Ref based on the provided input parameters.

Then, on the MATLAB Simulink it is required to design the VCIT-MPPT system. First it is required to complete the design as shown in the Figure 6.20. Then Python function block of the Py_MPPT should be coded.

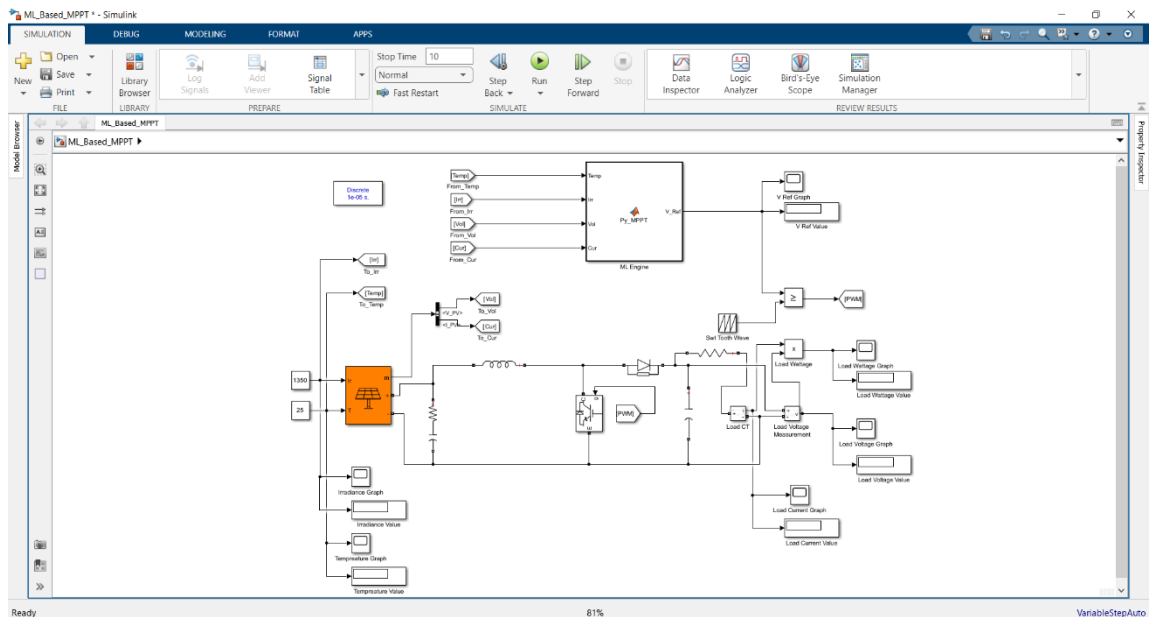
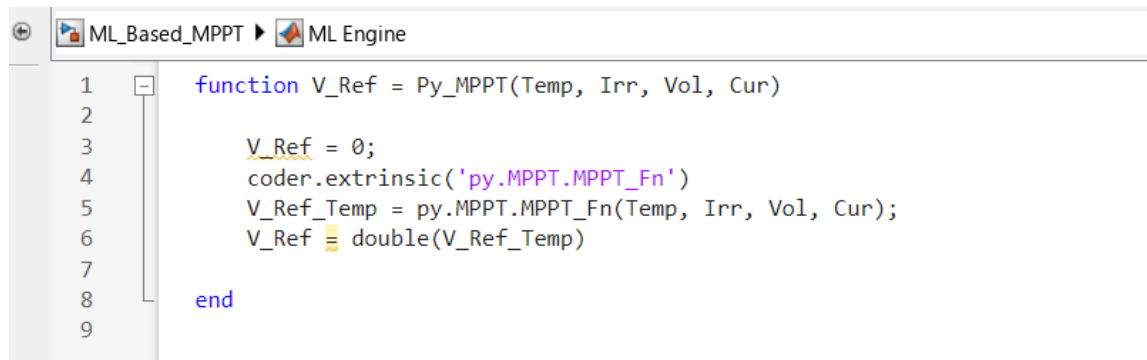


Figure 6.20: Designing of VCIT-MPPT system in MATLAB Simulink.

6.5 System Interfaces

Py_MPPT is a Python function block in Simulink offers a seamless integration of custom Python code within Simulink models, enabling to extend the functionality of their simulations. It allows users to incorporate Python scripts for custom computations, data processing, or interfacing with external systems. This block facilitates seamless integration of Python functionality into Simulink simulations with input and output ports

facilitating data exchange, it can execute Python code directly within Simulink environments.



```
ML_Based_MPPT ▶ ML Engine
1  function V_Ref = Py_MPPT(Temp, Irr, Vol, Cur)
2
3      V_Ref = 0;
4      coder.extrinsic('py.MPPT.MPPT_Fn')
5      V_Ref_Temp = py.MPPT.MPPT_Fn(Temp, Irr, Vol, Cur);
6      V_Ref = double(V_Ref_Temp)
7
8  end
9
```

Figure 6.21: Py_MPPT function block code.

6.6 Summary

The implementation of a machine learning-based VCIT-MPPT system utilized an Intel(R) Core (TM) i7-10510U CPU with 32 Giga Bytes of Random Access Memory running Windows 10 Pro (22H2). Essential software, including Python 3.11, Jupyter Notebook 6.4.12, MATLAB R2023b, and Simulink 23.2, were installed on this infrastructure. The VCIT-MPPT model employed Random Forest Regression, trained and evaluated using Python libraries like NumPy and scikit-learn. The model exhibited high accuracy (99.11%), low mean absolute error (0.0024), and minimal mean square error ($1.27e-05$), validating its reliability. The MPPT function, MPPT_Fn, integrated the Random Forest Regressor to predict voltage reference based on real-time environmental and electrical parameters. In Simulink, the Python Function block, Py_MPPT, enabled seamless execution of Python code within the simulation environment, extending functionality and facilitating data exchange between Simulink and Python.

CHAPTER 7

EVALUATION OF VCIT-MPPT SYSTEM

7.1 Introduction

Chapter 6 described about the implementation process of VCIT-MPPT system. This chapter describes about evaluation of the machine learning based VCIT-MPPT system. This chapter aims to assess the performance of the VCIT-MPPT system subjected various environmental conditions such as irradiance & temperature and to evaluate the model performance with the conventional MPPT system.

7.2 Selection of ML Model

Selection of the machine learning (ML) model for the VCIT-MPPT system is crucial for achieving accurate and efficient maximum power point tracking. Dataset analysis and visualization are the two most important parts prior selection of suitable machine learning model. As the first step missing values were checked and filled using forward fill (ffill) method and ensured there are no null values or missing values in the data set by using DataFrame.describe() function.

```
In [14]: data_set.describe()
```

```
Out[14]:
```

	Temp	Irr	Vol	Cur	V_Ref
count	165921.000000	165921.000000	165921.000000	165921.000000	165921.000000
mean	25.000000	1005.000000	298.656769	329.635181	0.657820
std	6.055319	228.082909	10.623255	73.581392	0.037850
min	15.000000	610.000000	264.426842	184.413604	0.547414
25%	20.000000	807.500000	291.176023	257.088700	0.619615
50%	25.000000	1005.000000	298.487809	322.822383	0.655686
75%	30.000000	1202.500000	306.255998	405.352663	0.699554
max	35.000000	1400.000000	326.863114	463.392800	0.714757

Figure 7.1: VCIT-MPPT dataset description using Jupyter Notebook

Then these data were normalized and visualized using seaborn pairplot() function as shown in Figure 7.2.

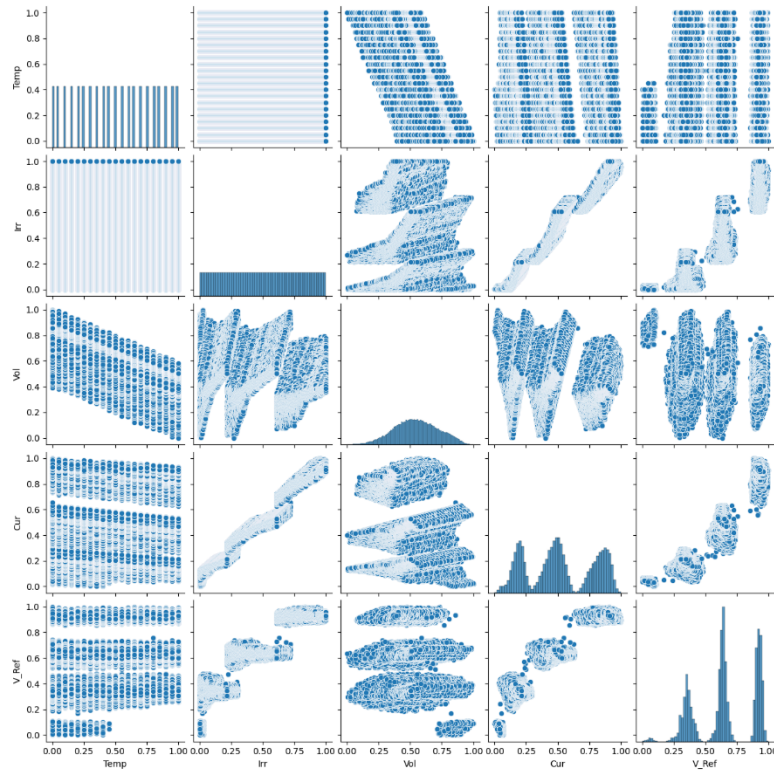


Figure 7.2: Visualization of VCIT-MPPT dataset using seaborn pairplot() function

Then these data were correlated and visualized using seaborn heatmap() function taken all the inputs as irradiance, temperature, voltage, current values and output as the reference voltage value as shown in Figure 7.3 and Figure 7.4.

```
In [51]: correlation = data_set.corr()['V_Ref']
In [52]: cor_rel = pd.DataFrame(correlation)
In [53]: cor_rel.columns = ['Correlation']
In [54]: cor_rel = cor_rel.sort_values(by = ['Correlation'], ascending = False)
In [55]: cor_rel
Out[55]:
```

	Correlation
V_Ref	1.000000
Cur	0.950184
Irr	0.921387
Temp	0.089818
Vol	-0.282672

Figure 7.3: Correlation of VCIT-MPPT dataset using pandas corr() function

Form the above correlation analysis it is understood that, voltage reference vs. current is having correlation coefficient of 0.950184. This implies that there is a strong positive correlation between voltage reference and current. This means that as voltage reference increases, current tends to increase as well, and vice versa.

Voltage reference vs. irradiance is having correlation coefficient of 0.921387. This implies that there is a strong positive correlation between voltage reference and irradiance. this indicates that as voltage reference increases, irradiance also tends to increase, and vice versa.

Voltage reference vs. temperature is having correlation coefficient of 0.089818. This implies that weak positive correlation is present between voltage reference and temperature. the correlation co-efficient being close to the value of zero suggests that there is little or there is no linear relationship between these two variables.

Voltage reference vs. voltage is having correlation coefficient of -0.282672. this implies that there is a moderate negative correlation between voltage reference and voltage. this means that as voltage reference increases, voltage tends to decrease, and vice versa.

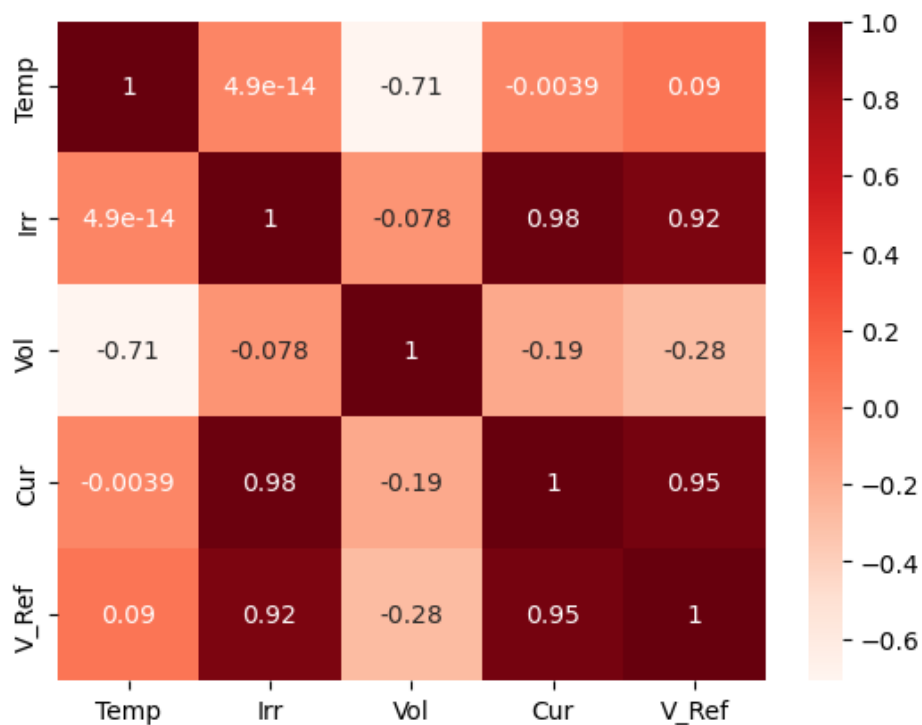


Figure 7.4: Visualization of VCIT-MPPT dataset using seaborn heatmap() function

From the heatmap it is evident that, current exhibits strong positive correlations with both irradiance (0.98) and voltage reference (0.95), indicating their simultaneous increase. However, its correlation with temperature is close to zero (-0.0039), implying minimal linear relationship, and it shows a moderate negative correlation with voltage (-0.19), which is implying an inverse relationship.

Voltage has a weak negative correlation with both voltage reference (-0.28) and current (-0.19), indicating a tendency to decrease as they increase. Additionally, there's a weak negative correlation with irradiance (-0.078) and a strong negative correlation with temperature (-0.71), implying an inverse relationship.

Temperature exhibits a weak positive correlation with voltage reference (0.09), while its correlations with irradiance and current are close to zero, suggesting minimal linear relationships. However, there's a strong negative correlation with voltage (-0.71), indicating a significant inverse relationship.

Irradiance shows strong positive correlations with both current (0.98) and voltage reference (0.92), indicating their simultaneous increase. However, its correlations with temperature and voltage are close to zero, suggesting minimal linear relationships.

Importantly there are strong positive correlations exist between voltage reference and both current (0.95) and irradiance (0.92), indicating that increases in voltage reference are associated with corresponding increases in current and irradiance. However, there's a weak positive correlation with temperature (0.09), suggesting a less pronounced relationship, and a moderate negative correlation with voltage (-0.28), implying an inverse relationship.

Random Forest Regression, Decision Tree Regression, Ridge Regression, Lasso Regression and Linear Regression model were defined, trained, evaluated and plotted the same using following Python codes in the Jupyter Notebook as shown in Figure 7.5.

```

In [74]: models = {'Linear Regression': LinearRegression(),
                  'Ridge Regression': Ridge(),
                  'Lasso Regression': Lasso(),
                  'Decision Tree Regression': DecisionTreeRegressor(),
                  'Random Forest Regression': RandomForestRegressor()
                }

In [75]: results = {}

In [76]: for name, model in models.items():
          model.fit(X_train, y_train)
          predicted = model.predict(X_test)
          mse = mean_squared_error(y_test, predicted)
          r2 = r2_score(y_test, predicted)
          results[name] = (mse, r2)

In [77]: fig, axes = plt.subplots(nrows=2, ncols=1, figsize=(10, 8))

          # MSE Plot
          axes[0].barh(range(len(results)), [mse for mse, _ in results.values()], color='skyblue')
          axes[0].set_yticks(range(len(results)))
          axes[0].set_yticklabels(results.keys())
          axes[0].set_xlabel('Mean Squared Error (MSE)')
          axes[0].set_title('Model Performance Comparison (MSE)')

          # R2 Plot
          axes[1].barh(range(len(results)), [r2 for _, r2 in results.values()], color='lightgreen')
          axes[1].set_yticks(range(len(results)))
          axes[1].set_yticklabels(results.keys())
          axes[1].set_xlabel('R-squared (R2)')
          axes[1].set_title('Model Performance Comparison (R2)')

          plt.tight_layout()
          plt.show()

```

Figure 7.5: Defining, training, evaluating and plotting various models.

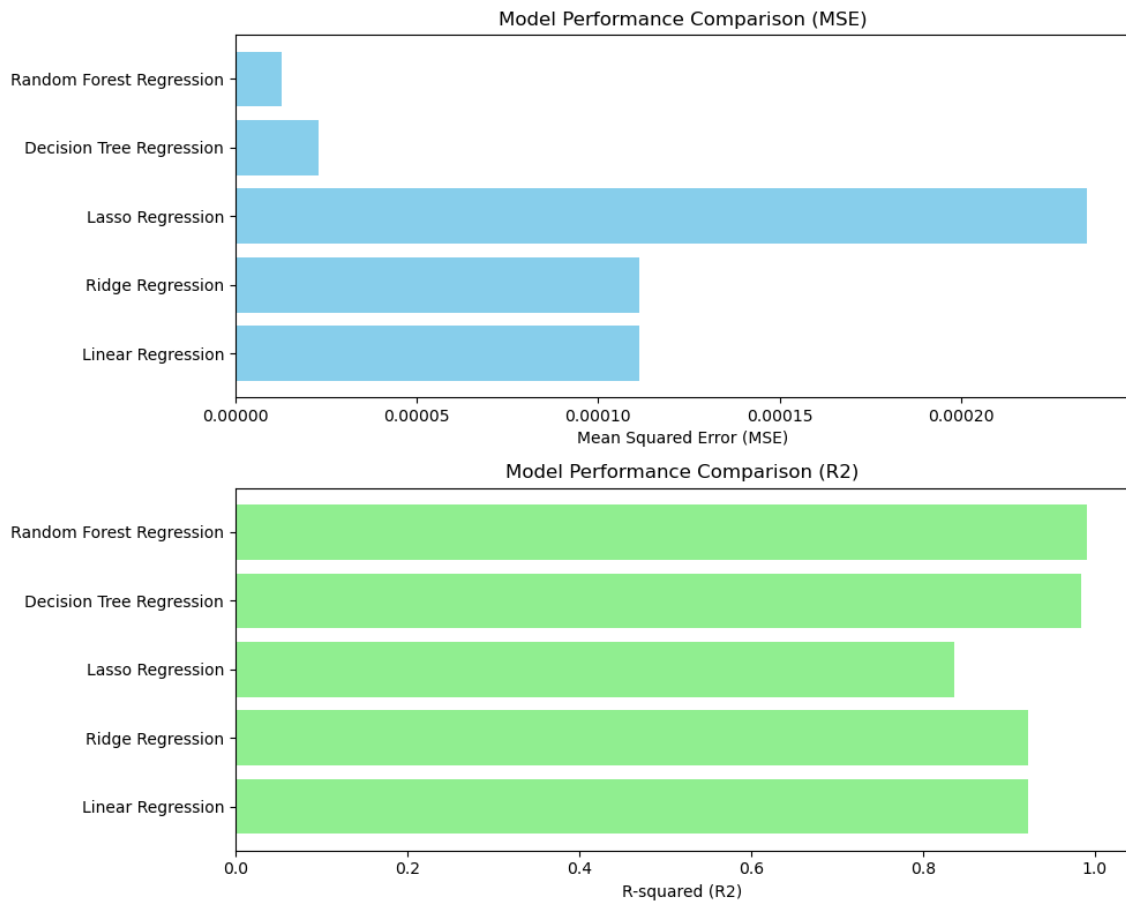


Figure 7.6: Selection of the machine learning (ML) model for the VCIT-MPPT system.

Variance refers to the model's sensitivity to fluctuations in the training data. High value of variance denotes that the model is overly sensitive to the training data and captures noise along with the underlying patterns. Conversely, a low variance suggests that the model is more robust and generalizes well to unseen data. Bias measures how much the predicted values differ from the true values. A high bias value suggests the model oversimplifies the data, missing its intricate patterns, resulting in underfitting. Conversely, a low bias indicates the model adeptly captures these patterns, closely fitting the training data without oversimplification.

Variance and bias were calculated and model has given 0.0014319905179881916 value as variance and 1.2697465569407419e-05 value as bias. These values suggest that the model has low variance and very low bias. This suggests that the model demonstrates stability (minimal fluctuation) and effectively grasps the fundamental patterns within the data (minimal distortion). As a result, it is expected to excel not only on the training dataset but also on new, unseen data, achieving a harmonious equilibrium between understanding intricate patterns and preventing overfitting. Figure 7.8 depicts a visualization of variance and bias concerning the true line.

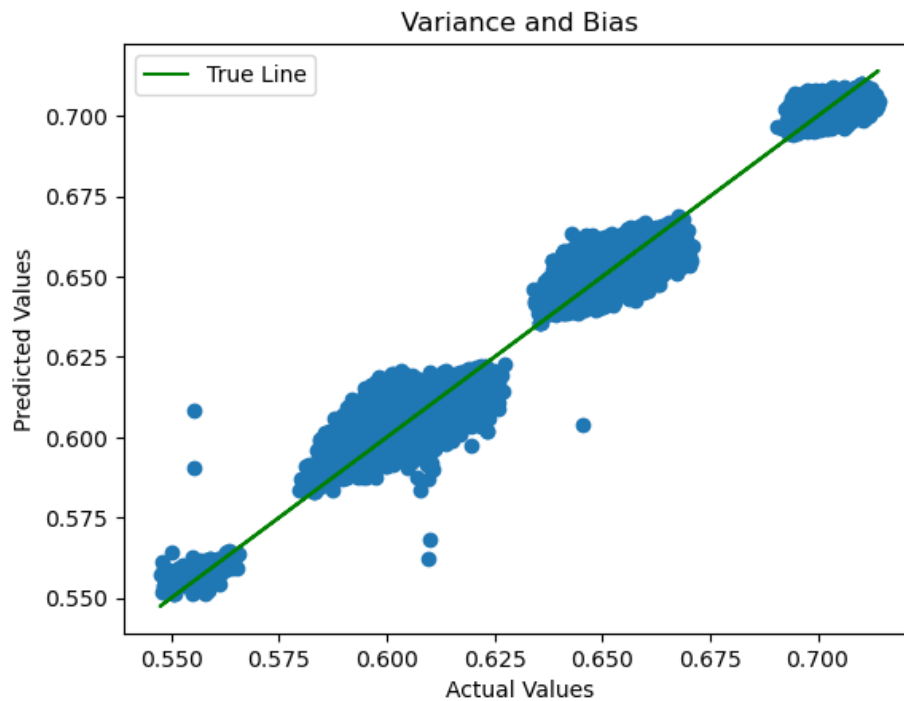


Figure 7.8: Variance and bias plot w.r.t. true line.

7.4 Dataset

The nature of the data employed encompasses several key parameters crucial for understanding and optimizing MPPT systems' performance. The data includes parameters related to the solar panel's electrical characteristics, such as instantaneous voltage and current drawn from the panel and varying environmental conditions temperature and incident irradiance levels. Ambient temperature data is incorporated into the datasets, as temperature variations significantly impact the efficiency and performance of solar panels. Temperature affects the electrical properties of the panel, including its voltage and current output, necessitating its consideration in MPPT system analysis. Irradiance values, representing the intensity of sunlight incident on the solar panel, which directly influences its power generation capability.

```
In [14]: data_set.describe()
```

Out[14]:

	Temp	Irr	Vol	Cur	V_Ref
count	165921.000000	165921.000000	165921.000000	165921.000000	165921.000000
mean	25.000000	1005.000000	298.656769	329.635181	0.657820
std	6.055319	228.082909	10.623255	73.581392	0.037850
min	15.000000	610.000000	264.426842	184.413604	0.547414
25%	20.000000	807.500000	291.176023	257.088700	0.619615
50%	25.000000	1005.000000	298.487809	322.822383	0.655686
75%	30.000000	1202.500000	306.255998	405.352663	0.699554
max	35.000000	1400.000000	326.863114	463.392800	0.714757

Figure 7.9: Description of Dataset

The dataset is able to capture the dynamic interaction between these parameters, reflecting real world conditions that MPPT systems operate within. By simulating these interactions using tools like MATLAB Simulink software, the impact on MPP under various scenarios can be studied. Furthermore, MATLAB Simulink can generate datasets in CSV format which can facilitate further analysis and processing.

7.5 Model Training

Before training the Random Forest Regression model, it's crucial to prepare the dataset. This involves steps such as handling missing values and splitting the dataset into training and testing sets to evaluate the model's performance.

```
In [63]: X = data_set[['Temp', 'Irr', 'Vol', 'Cur']]
         y = data_set['V_Ref']

In [65]: X_train, X_test, y_train, y_test = train_test_split(X, y, test_size = 0.3, random_state = 0)

In [66]: X_train
Out[66]:
```

	Temp	Irr	Vol	Cur
77930	24	1292.1	294.931232	430.043893
70038	23	1293.0	292.965168	436.820065
148921	33	1280.3	284.816520	424.572538
80717	25	780.7	311.867060	237.904505
143146	33	702.8	294.653736	228.047876
...
97639	27	892.7	300.880901	289.808823
96939	27	722.7	299.797076	238.940516
152315	34	829.6	278.468367	288.860647
117952	29	1343.8	294.762564	429.408789
43567	20	1016.2	306.992459	332.805966

116144 rows x 4 columns

```
In [67]: X_train.shape
Out[67]: (116144, 4)
```

Figure 7.10: Splitting the dataset into training and testing.

Dataset was split into two parts X and y. X subset of the dataset consists of the input parameters of the temperature, irradiance, voltage, current values and the y subset of the dataset consists of output parameter of reference voltage. X_train and X_test represent the training and testing datasets for features, while y_train and y_test correspond to the training and testing datasets for target variables. The data is split using a 70-30 ratio and random_state ensures reproducibility.

```

In [79]: regressor = RandomForestRegressor()
In [80]: regressor.fit(X_train, y_train)
Out[80]: RandomForestRegressor()
In [81]: predicted = regressor.predict(X_test)
In [82]: predicted.shape
Out[82]: (49777,)
In [83]: dataframe = pd.DataFrame(y_test, predicted)
In [84]: dfr = pd.DataFrame({'Actual V Ref': y_test, 'Predicted V Ref': predicted})
In [85]: dfr
Out[85]:

```

	Actual V Ref	Predicted V Ref
42423	0.657781	0.658984
97784	0.652105	0.656782
96434	0.610479	0.608037
99788	0.654063	0.654136
52417	0.655765	0.656089

Figure 7.11: Training the model using Random Forest Regression.

7.6 Testing of the Novel Design

The testing phase involves subjecting the VCIT-MPPT system to exposing the simulated environmental conditions to evaluate its functionality and robustness. This includes testing the system's ability to accurately track the MPP of solar panels under changing irradiance, temperature, voltage and current levels. Various test cases are designed to simulate different operating conditions and assess the system's response and performance.

During the testing procedure both conventional and Random Forest Regression based VCIT-MPPT system are exposed to same simulated environmental conditions and reference voltage (V_Ref) and the wattage of both systems were compared for following random environmental conditions.

Scenario 01 – Temp.25 °C & Irr. Level 1,000 W/m² (STC)

Scenario 02 – Temp.10 °C & Irr. Level 1,020 W/m²

Scenario 03 – Temp.15 °C & Irr. Level 600 W/m²

Scenario 04 – Temp.20 °C & Irr. Level 800 W/m²

Scenario 05 – Temp.22 °C & Irr. Level 1,100 W/m²

Scenario 06 – Temp.25 °C & Irr. Level 1,350 W/m²

Scenario 07 – Temp.30 °C & Irr. Level 1,250 W/m²

Scenario 08 – Temp.35 °C & Irr. Level 1,400 W/m²

Scenario 01 – Temp. 25 °C & Irr. Level 1000 W/m² (STC).

Conventional MPPT is giving values between 0.6501 – 0.6513 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 97,370 – 97,800 W. Figure 7.12 & 7.13 demonstrates the fluctuation of V_Ref and wattage.



Figure 7.12: Fluctuation of voltage reference in conventional MPPT at STC.

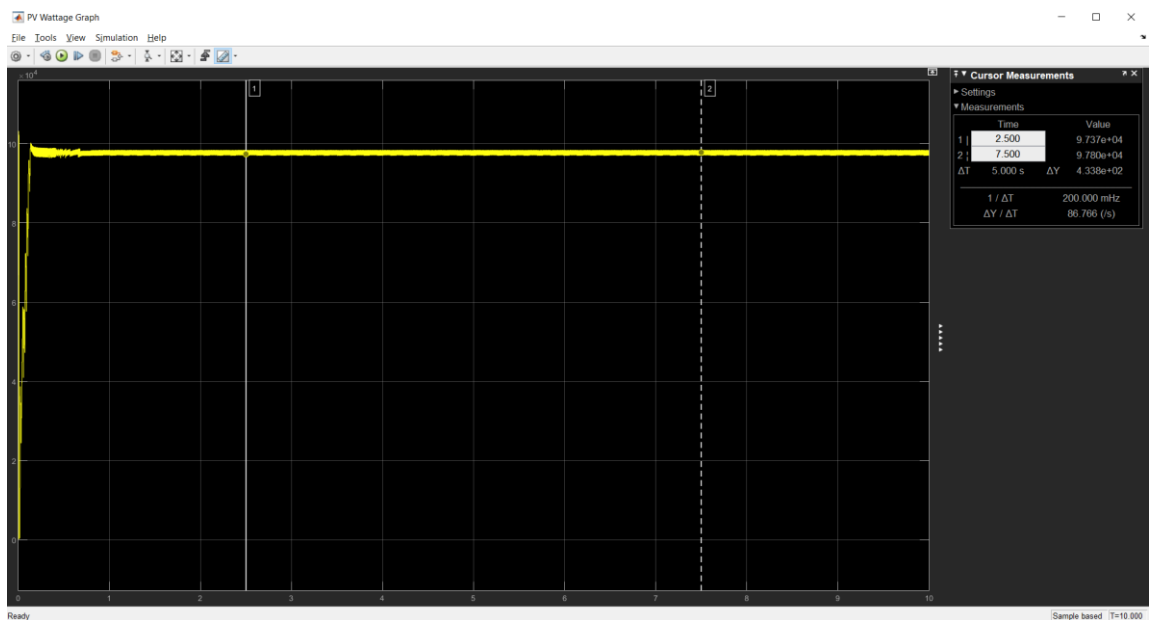


Figure 7.13: Fluctuation of wattage in conventional MPPT at STC.

However, VCIT-MPPT is giving value of 0.6514 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 99,190 W. Figure 7.14 & 7.15 demonstrates the steady V_Ref and wattage.

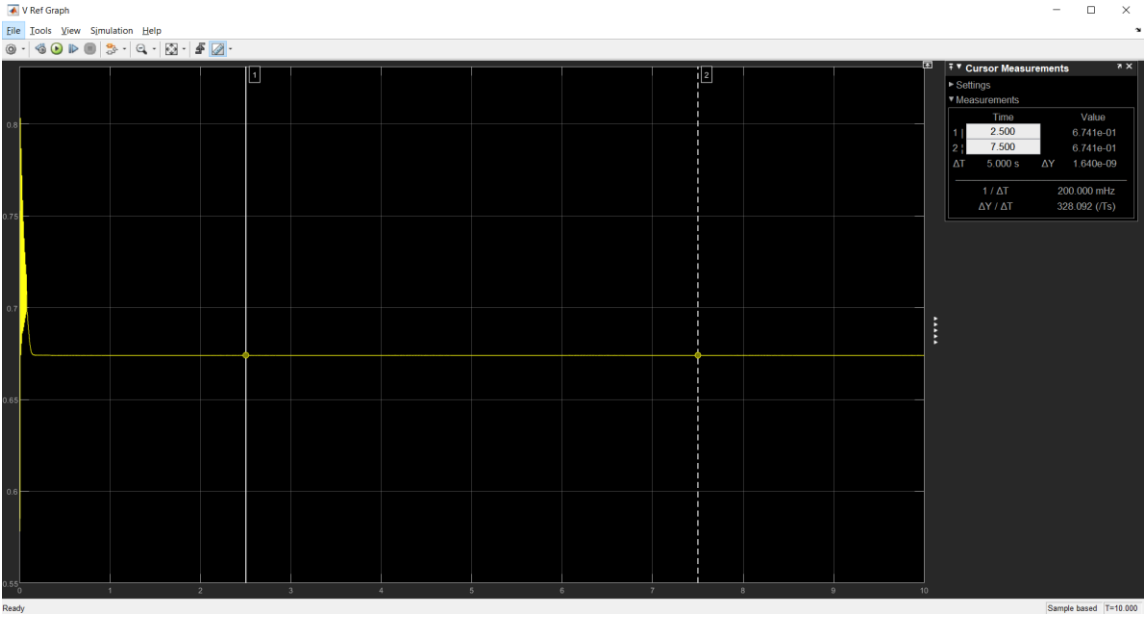


Figure 7.14: Voltage reference in VCIT-MPPT at STC.

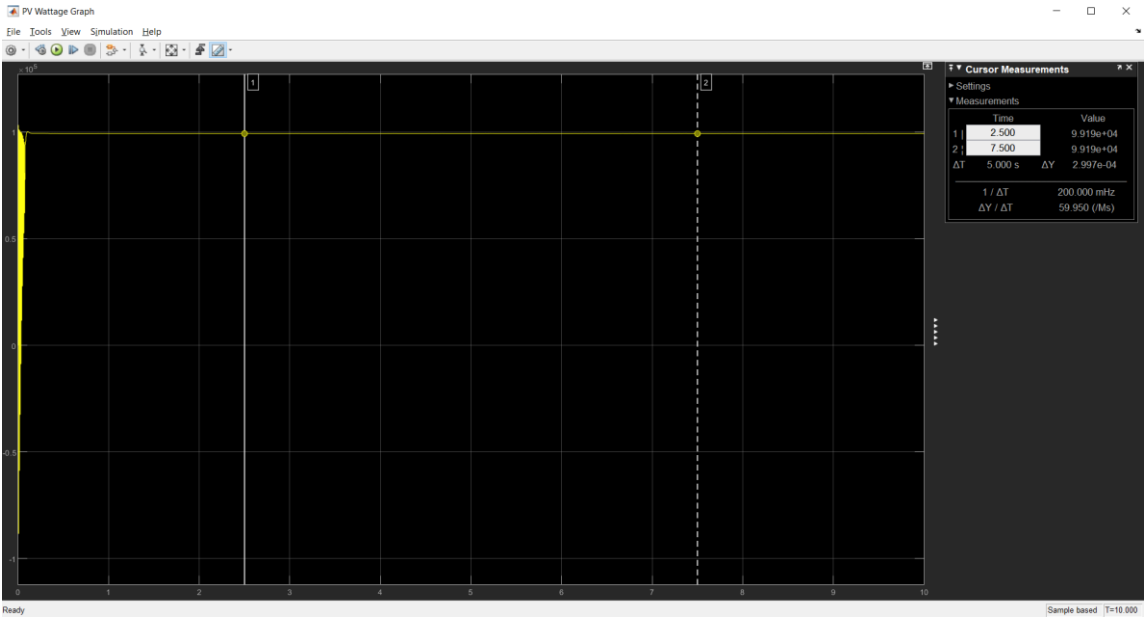


Figure 7.15: Wattage in VCIT-MPPT at STC.

Scenario 02 – Temp.10°C & Irr. Level 1,020 W/m².

Conventional MPPT is giving values between 0.6515 – 0.6595 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 106,300 – 107,300 W.

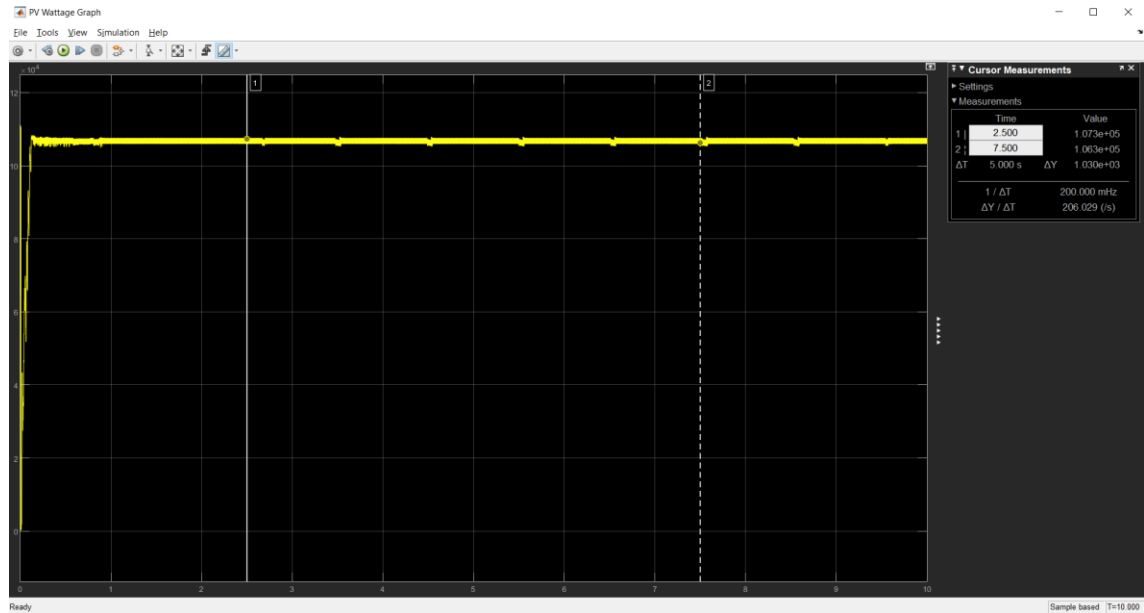


Figure 7.16: Fluctuation of wattage in conventional MPPT at Temp.10°C and & Irr. Level 1,020 W/m².

However, VCIT-MPPT is giving value of 0.6572 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 108,000 W.

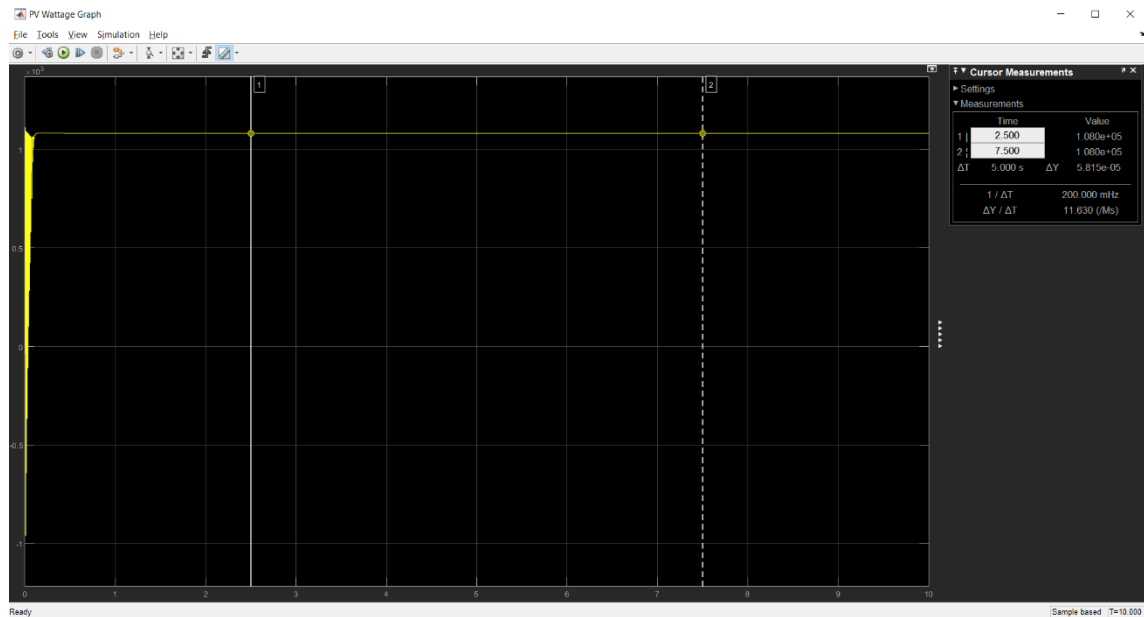


Figure 7.17: Wattage in VCIT-MPPT at Temp.10°C & Irr. Level 1,020 W/m².

Scenario 03 – Temp.15 °C & Irr. Level 600 W/m².

Conventional MPPT is giving values between 0.5525 – 0.5590 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 61,900 – 62,350 W.

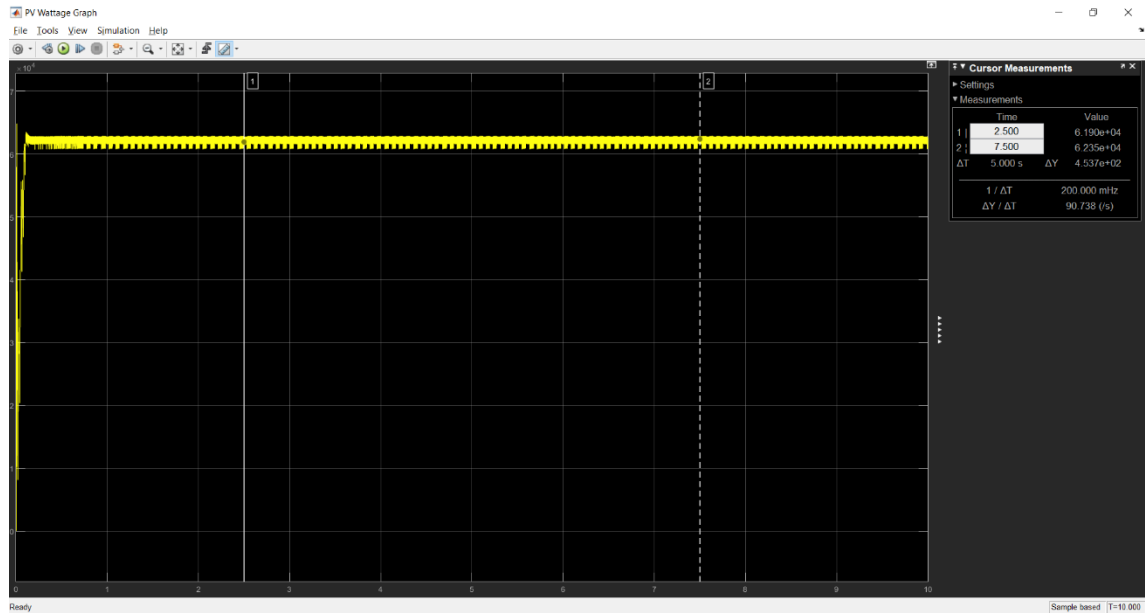


Figure 7.18: Fluctuation of wattage in conventional MPPT at Temp.15 °C & Irr. Level 600 W/m².

However, VCIT-MPPT is giving value of 0.5540 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 62,590 W.



Figure 7.19: Wattage in VCIT-MPPT at Temp.15 °C & Irr. Level 600 W/m².

Scenario 04 – Temp.20°C & Irr. Level 800 W/m².

Conventional MPPT is giving values between 0.5987 – 0.6118 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 77,990 – 80,020 W.

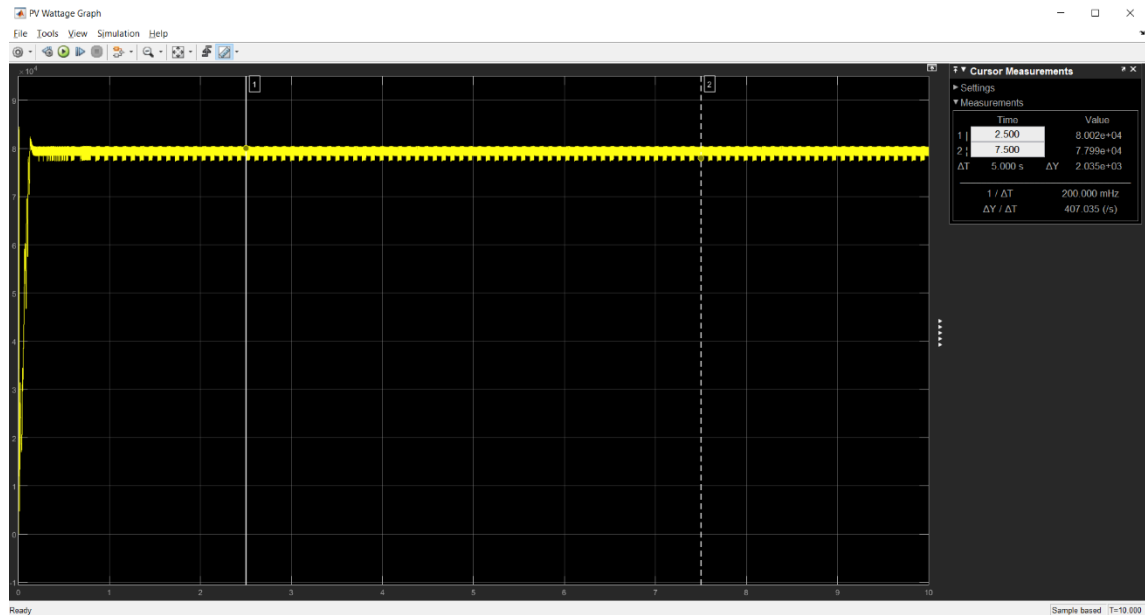


Figure 7.20: Fluctuation of wattage in conventional MPPT at Temp.20°C & Irr. Level 800 W/m².

However, VCIT-MPPT is giving value of 0.6086 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 80,120 W.



Figure 7.21: Wattage in VCIT-MPPT at Temp.20°C & Irr. Level 800 W/m².

Scenario 05 – Temp.22°C & Irr. Level 1,100 W/m².

Conventional MPPT is giving values between 0.6530 – 0.6533 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 105,600 – 106,000 W.

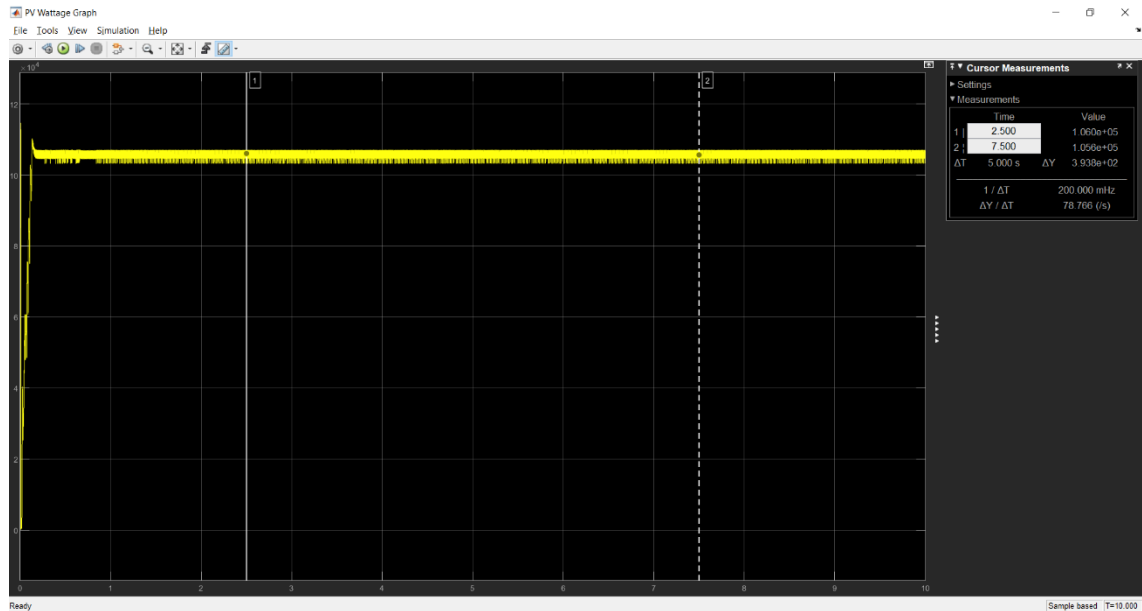


Figure 7.22: Fluctuation of wattage in conventional MPPT at Temp.22°C & Irr. Level 1,100 W/m².

However, VCIT-MPPT is giving value of 0.6530 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 106,800 W.



Figure 7.23: Wattage in VCIT-MPPT at Temp.22°C & Irr. Level 1,100 W/m².

Scenario 06 – Temp.25 °C & Irr. Level 1,350 W/m².

Conventional MPPT is giving values between 0.7014 – 0.7021 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 130,100 – 130,500 W.

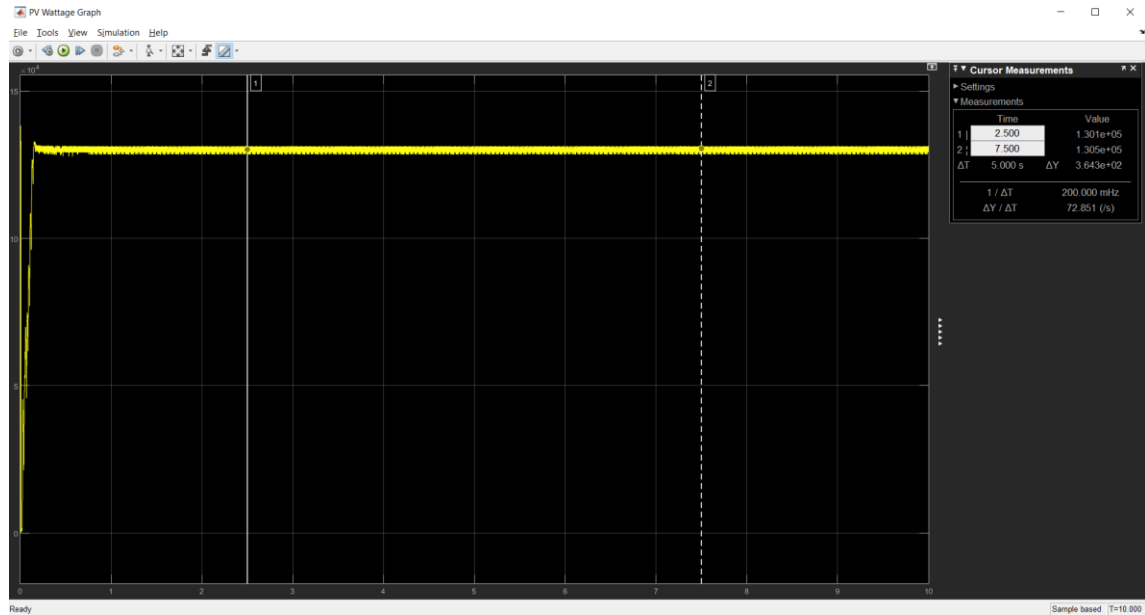


Figure 7.24: Fluctuation of wattage in conventional MPPT at Temp.25 °C & Irr. Level 1,350 W/m².

However, VCIT-MPPT is giving value of 0.6997 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 131,000 W.



Figure 7.25: Wattage in VCIT-MPPT at Temp.25 °C & Irr. Level 1,350 W/m².

Scenario 07 – Temp.30°C & Irr. Level 1,250 W/m².

Conventional MPPT is giving values between 0.6981 – 0.7011 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 120,300 – 120,500 W.



Figure 7.26: Fluctuation of wattage in conventional MPPT at Temp.30°C & Irr. Level 1,250 W/m².

However, VCIT-MPPT is giving value of 0.7219 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 120,600 W.

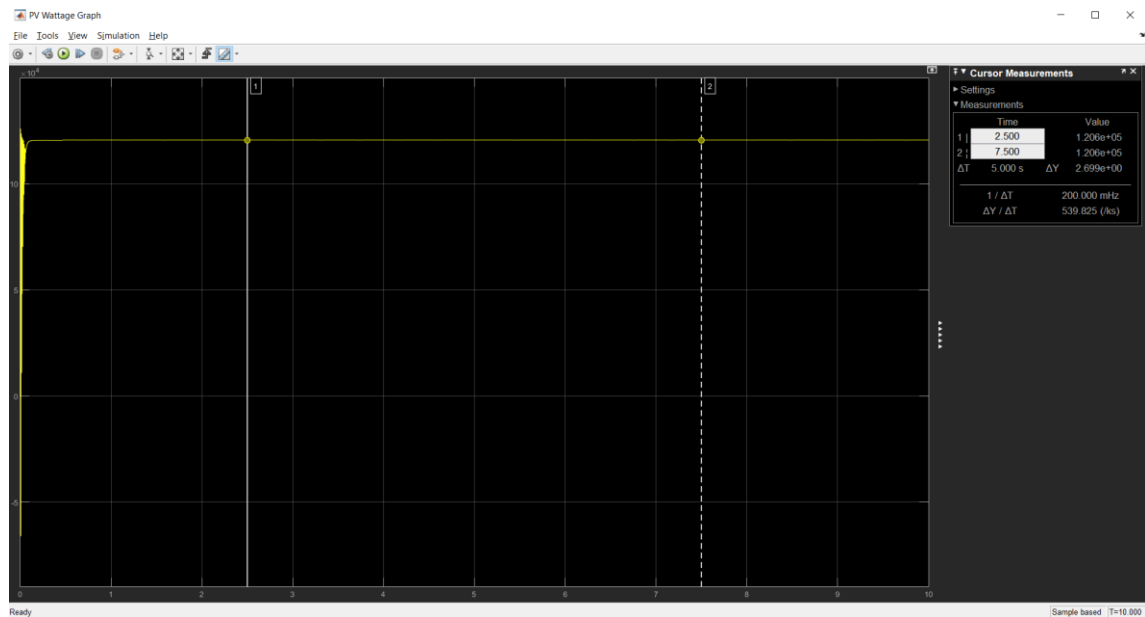


Figure 7.27: Wattage in VCIT-MPPT at Temp.30°C & Irr. Level 1,250 W/m².

Scenario 08 – Temp.35 °C & Irr. Level 1,400 W/m².

Conventional MPPT is giving values between 0.7007 – 0.7036 for voltage reference (V_Ref) and it is a fluctuating value which results fluctuation of system output power / wattage of the system as 127,000 – 127,400 W.

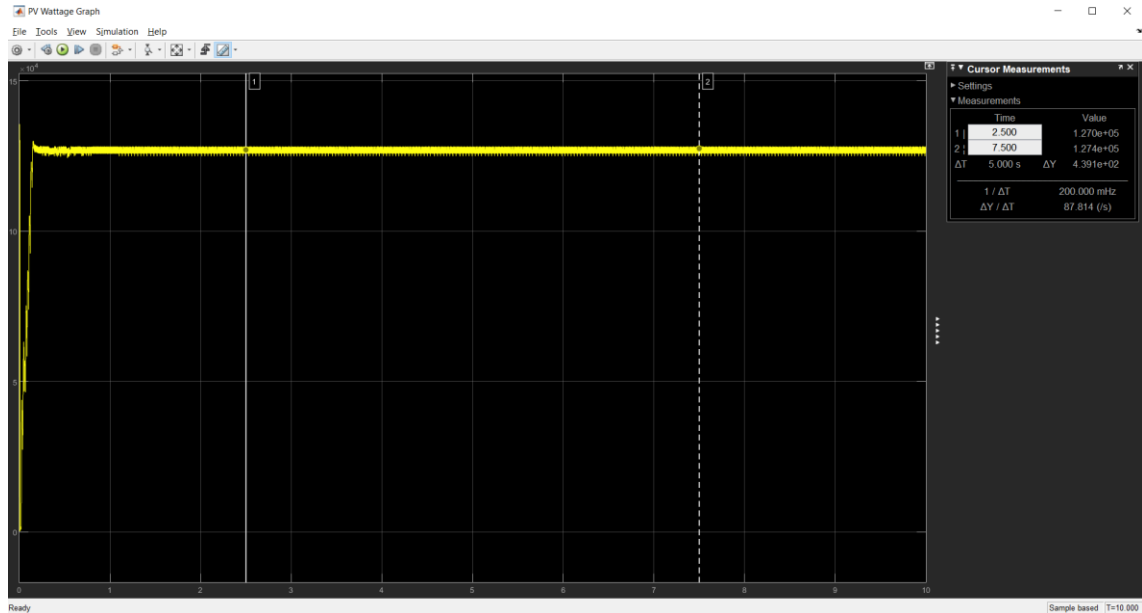


Figure 7.28: Fluctuation of wattage in conventional MPPT at Temp.35 °C & Irr. Level 1,400 W/m².

However, VCIT-MPPT is giving value of 0.7039 for voltage reference (V_Ref) and it is a steady value which results steady output power / wattage of the system as 129,000 W.

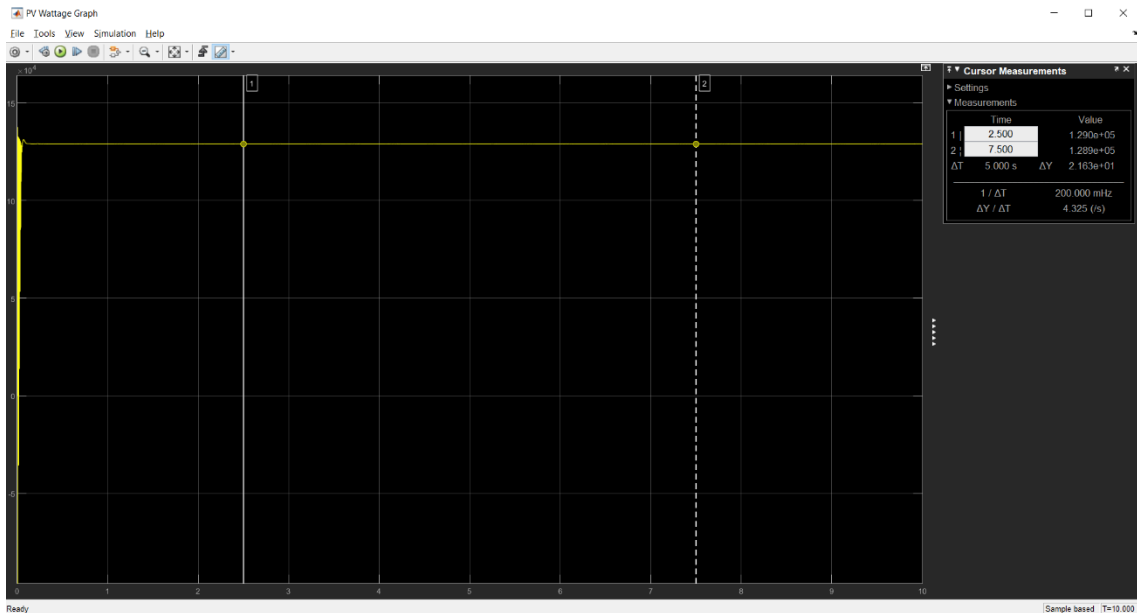


Figure 7.29: Wattage in VCIT-MPPT at Temp.35 °C & Irr. Level 1,400 W/m².

Results obtained during testing process were summarized and can be tabulated as follows: -

Sen.	Temp.	Irradiance	Conventional MPPT			VCIT-MPPT		
			V Ref	Wattage	Settling Time	V Ref	Wattage	Settling Time
01	25 °C	1,000 W/m ²	0.6501 – 0.6513	97,370 – 97,800 W	147 mS	0.6514	99,190 W	82 mS
02	10 °C	1,020 W/m ²	0.6515 – 0.6595	106,300 – 107,300 W	147 mS	0.6572	108,000 W	82 mS
03	15 °C	600 W/m ²	0.5525 – 0.5590	61,900 – 62,350 W	121 mS	0.5540	62,590 W	82 mS
04	20 °C	800 W/m ²	0.5987 – 0.6118	77,990 – 80,020 W	166 mS	0.6086	80,120 W	82 mS
05	22 °C	1,100 W/m ²	0.6530 – 0.6533	105,600 – 106,000 W	160 mS	0.6530	106,800 W	82 mS
06	25 °C	1,350 W/m ²	0.7014 – 0.7021	130,100 – 130,500 W	180 mS	0.6997	131,000 W	82 mS
07	30 °C	1,250 W/m ²	0.6981 – 0.7011	120,300 – 120,500 W	186 mS	0.7012	120,600 W	82 mS
08	35 °C	1,400 W/m ²	0.7007 – 0.7036	127,000 – 127,400 W	212 mS	0.7039	129,000 W	82 mS

Table 7.1: Summary of conventional MPPT vs VCIT-MPPT test results

7.7 Evaluation of the Novel Design

The evaluation phase involves analyzing the data collected during testing to assess the performance of the VCIT-MPPT system. Key metrics such as efficiency, accuracy, stability, and settling time are evaluated to determine how well the system performs compared to conventional MPPT methods.

Efficiency of the PV system is defined at the Standard Test Conditions (STC). As per the PV array configuration and technical data are shown in Figure 7.40 solar array consists of 47 x10 strings.

Array data	
Parallel strings	47
Series-connected modules per string	10
Module data	
Module:	User-defined
Maximum Power (W)	213.15
Cells per module (Ncell)	60
Open circuit voltage Voc (V)	36.3
Short-circuit current Isc (A)	7.84
Voltage at maximum power point Vmp (V)	29
Current at maximum power point Imp (A)	7.35

Figure 7.30 PV array configuration and technical data

Hence, maximum power output of the solar panel system can be calculated in following manner:-

$$\text{Voltage of the array at MPP} : 10 \times 29 = 290 \text{ VDC}$$

$$\text{Current of the array at MPP} : 47 \times 7.35 = 345.45 \text{ Amps}$$

$$\text{Maximum Power Output at MPP} : 290 \text{ VDC} \times 345.45 \text{ Amps} = 100,180.50 \text{ Watts}$$

Conventional MPPT is giving output wattage of 97,370 – 97,800 W which is 97.19 – 97.62% whereas VCIT-MPPT is giving output wattage of 99,190 W which is 97.19 – 99.01%. Hence, VCIT-MPPT has shown 1.39% efficiency improvement over conventional MPPT.

```
In [55]: print('Test Accuracy: ',regressor.score(X_test, y_test))
```

```
Test Accuracy: 0.991111697659911
```

```
In [56]: print('Train Accuracy: ',regressor.score(X_train, y_train))
```

```
Train Accuracy: 0.9987616468003666
```

Figure 7.31: Accuracy of the VCIT-MPPT model

The performance metrics provided for the VCIT-MPPT system utilizing Random Forest Regression are quite impressive. Model demonstrate training accuracy of 99.88% and testing accuracy of 99.11%. The train accuracy of 99.88% shows that the model fits the training data very well. It means that during the training phase, the model learned the patterns in the data almost perfectly. The test accuracy of 99.11% indicates that the model performs very well on unseen data. This means that when new data is introduced to the model, it can correctly classify or predict the outcome with high accuracy.

```
In [60]: print('Cross Validation Scores:',cross_val_score(regressor, X_train, y_train, cv=10))
```

```
Cross Validation Scores: [0.99091446 0.99120902 0.99088975 0.99160816 0.99091802 0.99061783  
0.99107612 0.99094008 0.99089089 0.99104953]
```

Figure 7.32: Cross Validation Scores of the VCIT-MPPT model

The cross validation scores offer further understanding of how the model performs across various data subsets. With scores ranging from around 99.08% to 99.16% displaying minimal variance and it indicates steady performance across different data folds. This suggests that the model demonstrates good generalization capabilities and avoids overfitting to the training data.

Hence, the system demonstrates a high level of precision in predicting the MPP of a PV system based on voltage, current, irradiance, and temperature inputs. This indicates that the model's predictions closely match the actual MPP values, which is critical for optimizing the energy output of PV panels.

Output Wattage of VCIT-MPPT system is stable whereas, in conventional MPPT system it was observed that, output wattage is varying by 430 – 2,030 Watts. Hence, VCIT-MPPT is showing significant stable output over conventional MPPT.

Settling time of the conventional MPPT system 121 – 212 milliseconds settling time and even after that also it is giving varying output whereas, in VCIT-MPPT system it is only taking 82 milliseconds settling time and giving a stable output.

7.8 Results

The evaluation outcomes offer valuable understanding into how well the VCIT-MPPT system performs and how effective it is. These findings might encompass visual data, statistical breakdowns, and performance indicators drawn from the testing period. Significant discoveries, like better energy capture, decreased power wastage, and enhanced dependability, are emphasized to showcase the advantages of this innovative approach.

Hence, these performances suggest that the VCIT-MPPT system utilizing Random Forest Regression is highly reliable and robust. It can effectively track the MPP of photovoltaic systems under varying environmental conditions, providing accurate predictions that can optimize energy harvesting and enhance the efficiency of solar power generation.

7.9 Summary

Assessing the VCIT-MPPT system's performance which utilizes machine learning methods to enhance MPPT efficiency, involves crucial steps such as selecting the right ML model and conducting thorough dataset analysis and visualization. Random Forest Regression emerges as the optimal choice due to its superior performance in terms of lower Mean Squared Error and higher R-squared value among the ML models considered for evaluation. Performance of the VCIT-MPPT system is evaluated, with a focus on testing its ability to accurately track the maximum power point under changing environmental conditions. The novel VCIT-MPPT system underwent rigorous testing to evaluate its functionality across simulated environmental conditions, focusing on accurately tracking solar panel power points. Compared to conventional MPPT, VCIT-

MPPT demonstrated steadier performance and higher efficiency. The evaluation phase analyzed data, emphasizing metrics like efficiency, accuracy, stability, and settling time. Results revealed VCIT-MPPT's superiority, showing 1.39% efficiency improvement over conventional methods. Utilizing Random Forest Regression, VCIT-MPPT achieved accuracy of 99.11% in predicting maximum power points, ensuring optimal energy output. Additionally, it exhibited stable output wattage and quicker settling times, highlighting its reliability and robustness and these findings emphasized VCIT-MPPT's effectiveness in optimizing solar power generation.

CHAPTER 8

CONCLUSION AND FURTHER WORK

The research has been carried out to enhance capabilities of Maximum Power Point Tracking (MPPT) in the solar power systems with less cost and computational power requirements. Objectives of the research which are critical review of evolution of MPPT techniques and in-depth study of technologies used in modern MPPT were covered in second chapter. Design and implement an ML model that utilizes inputs of solar panel voltage, current, irradiance, and temperature to optimize the performance of solar panel was covered in fifth chapter. Evaluation of the implemented solution is covered in the seventh chapter of this thesis. The novel approached VCIT-MPPT system aimed to optimize energy harvesting efficiency and reliability by accurately MPP of PV panels under varying environmental conditions. Through rigorous testing and evaluation, we have demonstrated the effectiveness and superiority of the VCIT-MPPT system compared to conventional MPPT methods.

The implementation process involved setting up a robust computing infrastructure and installing essential software components such as Python, Jupyter Notebook, MATLAB, and Simulink. Random Forest Regression was utilized for training the machine learning model, leveraging libraries like NumPy, pandas, and scikit-learn. The model exhibited impressive accuracy and minimal error metrics, ensuring reliable predictions of the MPP. During the testing phase, the VCIT-MPPT system was subjected to various simulated environmental conditions, including changes in temperature and irradiance levels. Comparative analysis with conventional MPPT systems revealed significant improvements in stability, efficiency, and settling time. The VCIT-MPPT system consistently provided steady voltage reference and output wattage, demonstrating its reliability and robustness across different scenarios.

One of the key findings of research was the efficiency improvement achieved by the VCIT-MPPT system. With an accuracy rate of 99.11% in predicting maximum power points, the system outperformed conventional methods, resulting in a 1.39% efficiency

improvement. This improvement is crucial for enhancing energy harvesting efficiency and reducing power losses in solar power systems, ultimately leading to increased renewable energy utilization. Moreover, the VCIT-MPPT system exhibited stable output wattage and quicker settling times compared to conventional MPPT systems. This stability is essential for maintaining consistent power output and optimizing energy generation in real-world applications. Additionally, the reduced settling time enhances the responsiveness of the system, allowing it to adapt quickly to changes in environmental conditions.

The results obtained from our evaluation provide compelling evidence of the effectiveness and reliability of the VCIT-MPPT system. By accurately tracking the MPP of photovoltaic panels under varying environmental conditions, VCIT-MPPT system gives significant benefits in terms of energy harvesting efficiency and reliability. These findings underscore the potential of machine learning-based approaches in optimizing solar power generation and advancing renewable energy technologies. One limitation identified when implementing the VCIT-MPPT on actual circuit will be usage of thermocouple and TVC (temperature voltage converter) unit for temperature measurements and irradiance sensor or photo-voltic sensor for irradiance level measurements will give millivoltage values and same needs to be amplified prior giving to the VCIT-MPPT. Hence, fresh dataset or mapping TVC and irradiance sensor / photo-voltic sensor for temperature and irradiance will be required prior training of the ML model since both sensors will be giving corresponding output values in terms of the voltage instead of temperature and irradiance.

In conclusion, the VCIT-MPPT system represents a significant advancement in MPPT technology, offering improved efficiency, stability, and responsiveness compared to conventional methods. The integration of machine learning algorithms enhances the system's predictive capabilities, enabling precise tracking of the MPP and maximizing energy output from solar panels with less cost and computational requirements.

REFERENCES

- [1] D. J. P. Kesari, Z. Haq, and A. Lather, "Solar Photovoltaic Power Generation," vol. 3, no. 7, 2021.
- [2] S. S. Shema, I. Daut, M. Irwanto, C. Shatri, N. Syafawati, and N. Ashbahani, "Study of inverter design and topologies for photovoltaic system," *International Conference on Electrical, Control and Computer Engineering 2011 (InECCE)*, 2011, Accessed: May 21, 2023. [Online]. Available: https://www.academia.edu/13092558/Study_of_Inverter_Design_and_Topologies_for_Photovoltaic_System
- [3] E. M. Vicente, P. dos Santos Vicente, R. L. Moreno, and E. R. Ribeiro, "High-efficiency MPPT method based on irradiance and temperature measurements," *IET Renewable Power Generation*, vol. 14, no. 6, pp. 986–995, 2020, doi: 10.1049/iet-rpg.2019.0849.
- [4] A. Luque and S. Hegedus, Eds., *Handbook of photovoltaic science and engineering*. Hoboken, NJ: Wiley, 2003.
- [5] M. L. Katche, A. B. Makokha, S. O. Zachary, and M. S. Adaramola, "A Comprehensive Review of Maximum Power Point Tracking (MPPT) Techniques Used in Solar PV Systems," *Energies*, vol. 16, no. 5, Art. no. 5, Jan. 2023, doi: 10.3390/en16052206.
- [6] B. Pozo, J. I. Garate, J. Á. Araujo, and S. Ferreiro, "Photovoltaic Energy Harvesting System Adapted for Different Environmental Operation Conditions: Analysis, Modeling, Simulation and Selection of Devices," *Sensors*, vol. 19, no. 7, Art. no. 7, Jan. 2019, doi: 10.3390/s19071578.
- [7] H. Sahraoui, H. Mellah, S. Drid, and L. Chrifi-Alaoui, "Adaptive maximum power point tracking using neural networks for a photovoltaic systems according grid," *Electrical Engineering & Electromechanics*, no. 5, pp. 57–66, Oct. 2021, doi: 10.20998/2074-272X.2021.5.08.
- [8] K. Y. Yap, C. R. Sarimuthu, and J. M.-Y. Lim, "Artificial Intelligence Based MPPT Techniques for Solar Power System: A review," *Journal of Modern Power Systems and Clean Energy*, vol. 8, no. 6, pp. 1043–1059, Nov. 2020, doi: 10.35833/MPCE.2020.000159.
- [9] "World Energy Transitions Outlook 2023: 1.5°C Pathway." Accessed: Nov. 24, 2023. [Online]. Available: <https://www.irena.org/Publications/2023/Jun/World-Energy-Transitions-Outlook-2023>
- [10] X. Ma *et al.*, "Data-Driven I–V Feature Extraction for Photovoltaic Modules," *IEEE Journal of Photovoltaics*, vol. 9, no. 5, pp. 1405–1412, Sep. 2019, doi: 10.1109/JPHOTOV.2019.2928477.

- [11] B. N. Nguyen, V. T. Nguyen, M. Q. Duong, K. H. Le, H. H. Nguyen, and A. T. Doan, "Propose a MPPT Algorithm Based on Thevenin Equivalent Circuit for Improving Photovoltaic System Operation," *Front. Energy Res.*, vol. 8, p. 14, Feb. 2020, doi: 10.3389/fenrg.2020.00014.
- [12] M. del Rosario, "A High Step up DC-DC converter with MPPT for PV application." Accessed: Nov. 24, 2023. [Online]. Available: <https://fardapaper.ir/mohavaha/uploads/2018/04/Fardapaper-A-high-step-up-DC-DC-converter-with-MPPT-for-PV-application.pdf>
- [13] J. Ramos-Hernanz, I. Uriarte, J. M. Lopez-Guede, U. Fernandez-Gamiz, A. Mesanza, and E. Zulueta, "Temperature based maximum power point tracking for photovoltaic modules," *Sci Rep*, vol. 10, no. 1, Art. no. 1, Jul. 2020, doi: 10.1038/s41598-020-69365-5.
- [14] M. Park and I.-K. Yu, "A study on the optimal voltage for MPPT obtained by surface temperature of solar cell," in *30th Annual Conference of IEEE Industrial Electronics Society, 2004. IECON 2004*, Nov. 2004, pp. 2040-2045 Vol. 3. doi: 10.1109/IECON.2004.1432110.
- [15] M. Khatri and A. Kumar, "Simulation and Experimental Validation of Hill-Climbing Algorithm for Maximum Power Point Tracking of Solar Photovoltaic Plant," *Current Science*, vol. 113, no. 07, Art. no. 07, Oct. 2017.
- [16] C. Li, Y. Chen, D. Zhou, J. Liu, and J. Zeng, "A High-Performance Adaptive Incremental Conductance MPPT Algorithm for Photovoltaic Systems," *Energies*, vol. 9, no. 4, Art. no. 4, Apr. 2016, doi: 10.3390/en9040288.
- [17] Y. Tan and D. s Kirschen, "Impact on the Power System of a Large Penetration of Photovoltaic Generation," presented at the IEEE Power Engineering Society General Meeting, Jul. 2007, pp. 1–8. doi: 10.1109/PES.2007.385563.
- [18] S. HADJI, J.-P. Gaubert, and F. Krim, "Maximum Power Point Tracking (MPPT) for Photovoltaic systems using open circuit voltage and short circuit current," presented at the 2013 3rd International Conference on Systems and Control, ICSC 2013, Oct. 2013. doi: 10.1109/ICoSC.2013.6750840.
- [19] S. B. Ch, J. Kumari, and R. Kullayappa, "Design and Analysis of Open Circuit Voltage Based Maximum Power Point Tracking for Photovoltaic System," *International Journal of Advances in Science and Technology*, vol. 2, pp. 51–60, Feb. 2011.
- [20] B. Ankaiah and N. Eswar, "Enhancement of Solar Photovoltaic Cell by Using Short-Circuit Current Mppt Method," vol. 2, pp. 45–50, Feb. 2013.
- [21] S.-P. Ye, Y.-H. Liu, C.-Y. Liu, K.-C. Ho, and Y.-F. Luo, "Artificial Neural Network Assisted Variable Step Size Incremental Conductance MPPT Method with Adaptive Scaling Factor," *Electronics*, vol. 11, no. 1, Art. no. 1, Jan. 2022, doi: 10.3390/electronics11010043.

- [22] M. T. Hussain, A. Sarwar, M. Tariq, S. Urooj, A. BaQais, and M. A. Hossain, "An Evaluation of ANN Algorithm Performance for MPPT Energy Harvesting in Solar PV Systems," *Sustainability*, vol. 15, no. 14, Art. no. 14, Jan. 2023, doi: 10.3390/su151411144.
- [23] B. Nayak, A. Mohapatra, and K. B. Mohanty, "Selection criteria of dc-dc converter and control variable for MPPT of PV system utilized in heating and cooking applications," *Cogent Engineering*, vol. 4, no. 1, p. 1363357, Jan. 2017, doi: 10.1080/23311916.2017.1363357.
- [24] Ž. Zečević and M. Rolevski, "Neural Network Approach to MPPT Control and Irradiance Estimation," *Applied Sciences*, vol. 10, no. 15, Art. no. 15, Jan. 2020, doi: 10.3390/app10155051.
- [25] K. J and F. Sy, "Modeling of a Photovoltaic Array in MATLAB Simulink and Maximum Power Point Tracking Using Neural Network," *J Electr Electron Syst*, vol. 07, no. 03, 2018, doi: 10.4172/2332-0796.1000263.
- [26] N. Mutoh, M. Ohno, and T. Inoue, "A Method for MPPT Control While Searching for Parameters Corresponding to Weather Conditions for PV Generation Systems," *IEEE Trans. Ind. Electron.*, vol. 53, no. 4, pp. 1055–1065, Jun. 2006, doi: 10.1109/TIE.2006.878328.
- [27] M. Tahruri *et al.*, "Maximum Power Point Tracking of PV System Based on Machine Learning," *Energies*, vol. 13, p. 692, Feb. 2020, doi: 10.3390/en13030692.
- [28] Department of Civil Engineering, E.T.S.I.C.C.P., Granada University. Spain, D. Gómez-Lorente, F. Aznar, and O. Rabaza, "MPPT algorithm based on Multiple Linear Regression model for Solar PV systems," *REPQJ*, vol. 20, pp. 100–105, Sep. 2022, doi: 10.24084/repqj20.233.
- [29] P. V. Mahesh, S. Meyyappan, and A. RamakoteswaraRao, "Maximum Power Point Tracking with Regression Machine Learning Algorithms for Solar PV systems," *International Journal of Renewable Energy Research (IJRER)*, vol. 12, no. 3, Art. no. 3, Sep. 2022.
- [30] H. Shareef, A. H. Mutlag, and A. Mohamed, "Random Forest-Based Approach for Maximum Power Point Tracking of Photovoltaic Systems Operating under Actual Environmental Conditions," *Computational Intelligence and Neuroscience*, vol. 2017, p. e1673864, Jun. 2017, doi: 10.1155/2017/1673864.
- [31] T. Sutikno, A. Samosir, R. Aprilianto, H. Purnama, W. Arsadiando, and S. Padmanaban, "Advanced DC-DC converter topologies for solar energy harvesting applications: a review," *Clean Energy*, vol. 7, pp. 555–570, May 2023, doi: 10.1093/ce/zkad003.

- [32] S. Manna *et al.*, “Design and implementation of a new adaptive MPPT controller for solar PV systems,” *Energy Reports*, vol. 9, pp. 1818–1829, Dec. 2023, doi: 10.1016/j.egy.2022.12.152.
- [33] R. Sharmin, S. S. Chowdhury, F. Abedin, and K. M. Rahman, “Implementation of MPPT Technique of Solar Module with Supervised Machine Learning”.
- [34] H. A. Behabtu *et al.*, “A Review of Energy Storage Technologies’ Application Potentials in Renewable Energy Sources Grid Integration,” *Sustainability*, vol. 12, no. 24, Art. no. 24, Jan. 2020, doi: 10.3390/su122410511.
- [35] Y. Li *et al.*, “Optimal battery schedule for grid-connected photovoltaic-battery systems of office buildings based on a dynamic programming algorithm,” *Journal of Energy Storage*, vol. 50, p. 104557, Jun. 2022, doi: 10.1016/j.est.2022.104557.
- [36] J. Pande, P. Nasikkar, K. Kotecha, and V. Varadarajan, “A Review of Maximum Power Point Tracking Algorithms for Wind Energy Conversion Systems,” *Journal of Marine Science and Engineering*, vol. 9, no. 11, Art. no. 11, Nov. 2021, doi: 10.3390/jmse9111187.