People's Democratic Republic of Algeria
Ministry of Higher Education and Scientific Research

University M'Hamed BOUGARA – Boumerdès

Institute of Electrical and Electronic Engineering

Department of Power and Control

Final Year Project Report Presented in Partial Fulfilment of the Requirements of the Degree of

'MASTER'

In **Control**

Option: **Control**

Title:

Development of New Maximum Power Tracking Techniques for Stand-alone PV System under Non-uniform Irradiance Conditions

Presented By:

- **BELMADANI Hamza**
- **MELLAL Sohaib**

Supervisor:

Prof. KHELDOUN Aissa

ABSTRACT

The overwhelming need to decarbonize the energy sector to peter out climate changes, and catch up with the increasing demand of energy, have paved the way to an immense deployment of renewables around the globe.

Solar systems are used to convert sunlight that hits their panels into electrical energy via the photovoltaic effect. However, photovoltaics have a very low efficiency, and the generated power depends almost entirely on the amount of collected solar irradiance, temperature, the electrical load and the ambient circumstances that surrounds them. Since it is not possible to have a fixed stream of solar radiation or temperature, it is crucial to come up with effective means to tackle these problems. In this regard, Maximum power trackers are integrated with PV systems to cope with the dynamically fluctuating operating conditions, and keep the generated power as high as possible.

This thesis focuses on maximum power point tracking (MPPT) in PV systems using soft computing techniques. Equilibrium Optimizer, Seagull Optimization and Slime Mould Algorithm are three novel metaheuristic techniques proposed in this project. Matlab and Simulink are used to simulate a standalone PV system driven by an MPPT controller and assess the three stated optimizers.

The recommended techniques demonstrated outstanding results, under distinct insolation levels and complex shading conditions. To confirm their effectiveness, a comparative study on the basis of robustness, convergence time and efficiency, is carried out along with other well-known techniques: Particle Swarm Optimization (PSO), Whale Optimization (WOA), Grey wolf Optimization (GWO), Wind Driven Optimization (WDO) and the Grasshopper Optimization algorithm (GOA). Obtained results revealed that the proposed algorithms are either superlative or competitive in terms of both convergence speed and tracking efficiency.

DEDICATION

I dedicate this dissertation to

The sake of Allah, my Creator and my Master.

My great and beloved parents, without all their prayers, sacrifices and blessings throughout all my life stations, bleak times, and tough periods, none of my success would be possible.

To the memories of my uncles: Smail, Nassr Eddine and Mohammed who passed away recently. They have left a void never to be filled in our lives and have meant and continue to mean so much to me. May Allah have mercy on them.

To the 'Hirak' militants who are striving for a state of justice, freedoms and dignity.

Hamza BELMADANI

This dissertation is dedicated to

My beloved parents and my brother, who where my source of motivation all the way through, who raised me to become the person I am today. You have been with me every step of the way, through good times and bad. Thank you for all the unconditional love, guidance, and support that you have given me. I wouldn't have made it without you. Thank you for everything.

To my family and my friends who helped me in every way possible, thank you for your being by my side.

Sohaib MELLAL

ACKNOWLEDGMENT

In the name of Allah the Most Beneficent and the Most Merciful, we thank Allah for all His blessing and strength that He gave us in completing this modest project.

We would like to express our deepest and sincere gratitude to our project Supervisor ***Professor Aissa Kheldoun*** *for his continuous assistance, guidance and patience. His spacious knowledge on the subject, was paramount to the realization and completeness of this project.*

Hamza BELMADANI– Sohaib MELLAL

Words cannot express enough my warm gratefulness to ***Professor Reda Deghbar*** *for his immense support and motivation, especially during my mother's illness. I am very honoured and fortunate that I met such a great person.*

I also would like to express my wholehearted thanks to my family for their generous support they provided for me and my parents throughout this tough period.

From the bottom of my heart, I would like to express my profound gratitude for my wonderful friends and cousins: Khaled Rezgui, Adel Henna, Karima Boulekraouet, Ahmed Bourouis, Moussa Lamrani, Lydia Djebari, Marwa Belatrech and Dounia Allali for their genuine support, motivation and push for tenacity.

Hamza BELMADANI

Table of Contents

List of Figures :

List of Tables

List of Acronyms

A	Diode quality factor
AC	Alternative Current
BoS	Balance of System
DC	Direct Current
DSP	Digital Signal Processor
EO	Equilibrium Optimizer
FF	Fill factor
FPGA	Field Programmable Gate Array
G	Irradiance

GOA	Grasshopper Optimization
GWO	Grey Wolf Optimization
GMPP	Global Maximum Power Point
I	Current
IC	Incremental Conductance
I_{ph}	Photo current
I_{MPP}	Maximum Power Point Current
I_{sh}	Short Circuit Current
I_{PV}	Photovoltaic Current
K	Boltzman constant
LMPP	Local Maximum Power Point
MPP	Maximum Power Point
MPPT	Maximum Power Point Tracking
MPA	Marine Predators Algorithm
OP	Operation Point
P	Power
P_{max}	Maximum Power
P_{pv}	Photovoltaic Module Power
PWM	Pulse Width Modulation
P&O	Perturb And Observe

PSO	Particle Swarm Optimization
Q	Charge of electron
R_{sh}	Shunt resistance
R_s	Series resistance
SOA	Seagull Optimization Algorithm
SMA	Slime Mould Algorithm
T	Temperature
V	Voltage
V_{MPP}	Maximum Power Point Voltage
V_{PV}	Photovoltaic Module Power
V_{oc}	Open Circuit Voltage
WDO	Wind Driven Optimization

GENERAL INTRODUCTION

With the rising efforts steered towards palliating gas emissions and ensuring a clean energy future, along with the significant and continuous decline in photovoltaic module prices, installation and equipment costs, solar energy is becoming more and more prevalent around the globe than ever, and increasingly competing with fossil fuels and polluting energy sources.

Another motivation that greased the wheels to a growing uptake of solar energy systems, is the emerging promising and milestone technologies that led to potential improvements to the solar cell efficiency. Nonetheless, the fluctuating nature of the generated power caused by changing atmospheric conditions throughout the day, leads to significant losses and hence a poor power conversion efficiency. Consequently, dynamic monitoring is required to detect unpredictable changes in weather conditions, and accordingly drive the operating point of the system efficiently to fully exploit the highest possible power from PV panels. The technique by which, the system is supervised and controlled to benefit from the available solar energy, to the highest extend is called Maximum Power Point Tracking (MPPT). This solution is highly cost effective and reduces the complexity of the system, instead of adding more panels and equipment which requires more area to be occupied, and increases the costs substantially.

The primary mission assigned to MPP trackers is to impel the operating point of the system towards its optimum point at which the power is maximum, for whatever weather conditions. This point is the peak of the nonlinear current-voltage characteristic curve of the PV panel, and it is mainly affected by the surrounding circumstances, precisely: Solar irradiance and temperature. MPPT controllers use different strategies to drive the system efficiently, classical techniques, such as Perturb and observe (P&O) and Incremental conductance (IC), are widely popular due to their simplicity and ease of implementation. However, these techniques cannot handle cases at which the PV array is subjected to non-uniform solar irradiance, this situation is known as Partial Shading Conditions (PSC) and it occurs when parts of the PV array are shaded due to certain external influences.

This urged the research community to propose an enormous number of soft computing and artificial intelligence techniques to tackle the effects of partial shading conditions. Most of them have demonstrated effectiveness in dealing with various situations, and they far outperform conventional methods. However, these optimizers have distinct characteristics concerning their convergence speed, robustness, efficiency and implementation complexity, and may sometimes fail under complex shading patterns. This impelled us to investigate the latest progress in the world of soft computing and metaheuristic algorithms. A survey of some prominent and recent techniques is provided, along with their feasibility in Maximum Power Point Tracking.

The main contribution of this project is to introduce the implementation of three powerful and fast novel metaheuristic algorithms: Equilibrium Optimizer (EO), Seagull Optimization Algorithm (SOA) and Slime Mould Algorithm (SMA), in maximum power tracking, to drive a standalone PV system exposed to several solar irradiance levels and shading scenarios. Tracking speed and

efficiency collected from various simulation tests, are the assessment parameters used to carry out the performance evaluation of the proposed optimizers.

The remainder of this report is organized as follows:

Chapter 1 provides an introduction to photovoltaic systems, their types and main components as well as the effects of ambient solar irradiance and temperature on the characteristics of solar panels.

Chapter 2 steps through the essentials of MPPT theory and provides a quick review of the most widely used classical techniques. Their characteristics are then deduced from a Matlab simulation.

In Chapter 3, several metaheuristic algorithms as well as three recent nature inspired optimizers are surveyed, along with their operating flowcharts in maximum power point tracking. A performance evaluation of these algorithms in a standalone PV system and a comparative study are carried out in Chapter 4.

The report is culminated with a general conclusion including proposals for future work.

CHAPTER 1 : Review of Photovoltaic Systems

1.1 Introduction:

Photovoltaic systems are made up of one or more solar panels that are composed of smaller units called solar cells. The latter play the central role in collecting solar energy and convert it into unregulated electrical energy.

Besides from solar panels, PV systems generally consist of multiple individual components and equipment, that are required for electricity conversion, regulation, distribution and storage, including, mechanical and electrical connections and mountings. Cumulatively, the collection of these elements form what is named the Balance of System usually denoted by (BoS). The design of the BoS is crucial for the system to operate efficiently and guarantee safe delivery of power to the load.

A typical PV system consists of the following key elements:

- **PV modules:** Or PV panels, are the backbone of the PV system as they are responsible for DC electricity generation from sunlight. A collection of connected PV modules form a solar array.
- **DC-DC Converters:** The output DC voltage from the PV panels can be boosted or stepped-down to the required level using DC-DC converters.
- **Battery Banks**: Due to the intermittent nature of the generated power from PV panels, battery banks are used to store the excess electric energy, and make sure that the system is able to supply electricity when solar energy is insufficient (during the night or during cloudy climate conditions). The three types of batteries that are most common to RE systems are:

  - Flooded Lead-Acid Batteries (FLA).
  - Sealed Absorbed Glass Mat Batteries (AGM).
  - Sealed Gel Cell Batteries.

  In general, the battery energy storage is required to meet the following requirements: low self-discharge rate, strong charging–discharging capability, easy maintenance, long lifespan, low price, and large range of working temperature [2].

- **Charge Controllers**: Charge controllers are used whenever a battery bank is included in the PV system. The main purpose of these devices is to protect and prevent batteries from overcharging or excessive discharging, by regulating the amount of current taken from or delivered to them. The two main types are PWM (Pulse Width Modulated) and MPPT (Tracking). PWM technology is older and more commonly used on smaller solar arrays. MPPT charge controllers can track the maximum power point of a solar array and deliver 10-25% more power than a PWM controller could do for the same array. They do that by converting excess voltage into usable current [1].

- **Inverters:** Inverters are used to convert DC power from the PV modules or the storage devices into AC power.
- **Transformer:** A transformer is used when as step-up/down action for the output AC voltage from the inverter is required to meet the desired level for the given operation.

1.2 Types of PV systems:

There exist two main configurations of solar systems:

1.2.1 Standalone PV systems:

Standalone Systems are those that rely solely on the solar energy and operate independently from the electric utility grid. They generally incorporate a bank of batteries to provide backup power at night or during emergency situations at which solar energy is insufficient to catch up with the load demands. A general scheme of a typical standalone system is depicted in figure 1.

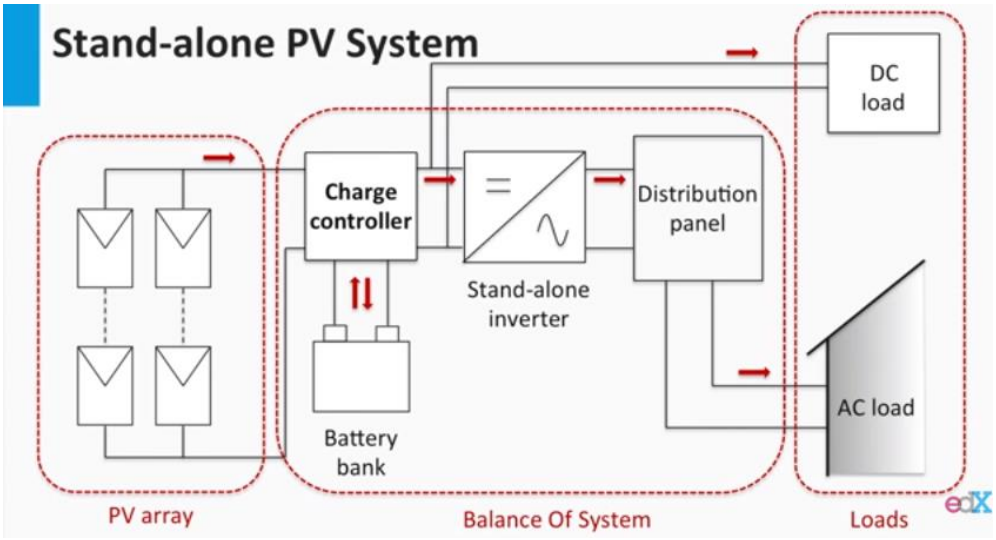

Figure 1: General Scheme of a Typical Standalone System [3]

1.2.2 Grid-Connected PV systems:

As the name indicates, in grid connected systems, the PV system AC output circuits are linked to the electric utility grid. This allows exchanging power between the two sides depending on the circumstances. In many situations, the PV system provides a much higher power than what the load requires, at that time the excess power is fed to the grid. If on the other hand, the power generated by the solar system is not sufficient, then the balance amount required will be delivered by the utility grid. A general scheme of a typical grid-connected PV system is depicted in figure 2.

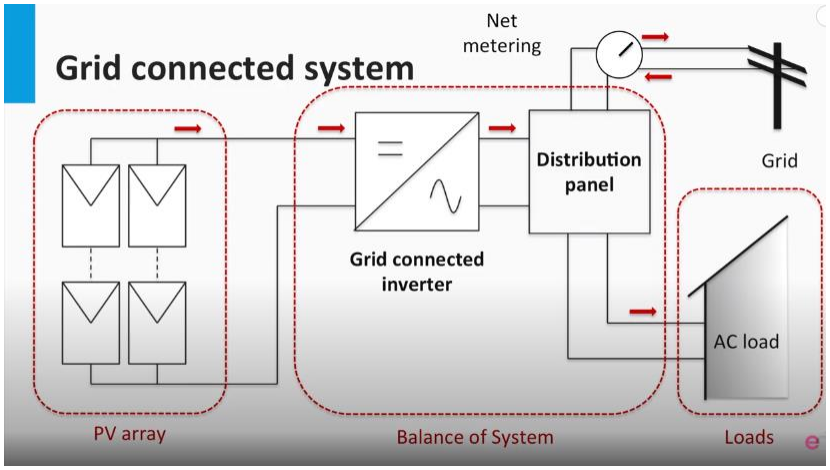

Figure 2: General Scheme of a Typical Grid-Connected PV System [2]

1.3 Advantages and Disadvantages of PV systems:

1.3.1 Advantages:

- Clean and silent energy without any noise pollution.
- Environmentally friendly.
- Solar energy can be made available almost anywhere there is sunlight.
- Photovoltaic panels, through photoelectric phenomenon, produce electricity in a direct electricity generation way.
- The PV system can be constructed to any size based on the power requirements of the consumers.
- Economically viable, since Solar panels cost is currently on a fast reducing track and is expected to continue reducing for the next years.
- Provide an effective solution to energy demand during peak periods.
- Low operating and maintenance costs comparing to other energy systems.
- No mechanically moving parts, thus it requires less maintenance.

1.3.2 Disadvantages:

- Solar resources dependent, no power output at night and also during daytime on a cloudy or rainy weather.
- Utilization of toxic chemicals during the production of the PV panels.
- Less reliable solution because of the intermittency and unpredictability of solar energy.
- Require additional equipment (inverters) to convert direct electricity (DC) to alternating electricity (AC) and storage batteries, in order to be used on the power network, thus increasing the investment cost for PV panels considerably.

- Requires a lot of space.
- Solar panels efficiency levels are relatively low (between 14%-25%) compared to the efficiency levels of other energy systems.
- PV systems are fragile and can easily be damaged.

1.4 PV Cells, Modules and Arrays:

PV cells are the building blocks of solar panels and the core elements that absorb incident light and convert it into electric energy. A solar cell is basically a specially designed semiconductor device (Usually made of Silicon) whose PN junction is exposed to sunlight. When solar radiation strikes the junction, electrons are released from the atoms of the semiconductor material, which causes an electric current to flow through, in a process called the photovoltaic effect.

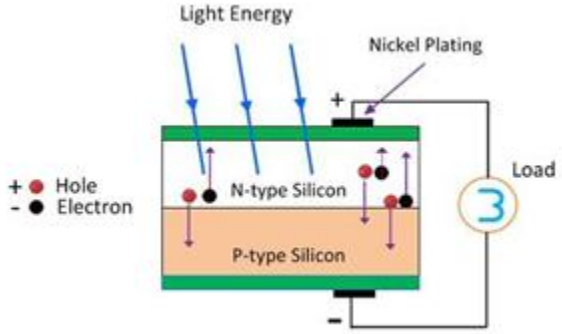

Figure 3: Construction of a Silicon based PV Cell [4]

Since an individual PV cell produces a small amount of power (1 or 2W), several PV cells are stringed together in series (for higher voltage) and in parallel (for higher current) forming a PV module to obtain the desired output level. In the market the maximum power capacity of the module is 1 kW, even though higher capacity is possible to manufacture, it will become cumbersome to handle more than 1 kW module [5]. At that time in a similar manner, PV modules can be wired together in series and in parallel to form a solar array, when the power generated by a single panel is not sufficient to meet the load requirements.

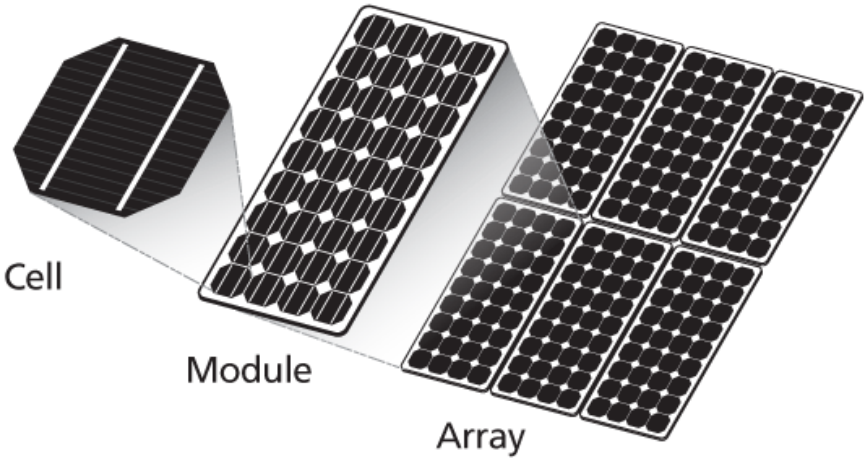

Figure 4: PV cell, Module and Array Arrangement.

1.5 PV Modelling:

With different degrees of accuracy and complexity, a number of electronic circuits have been proposed in literature to model the PV cell. An ideal solar cell can be modeled using a photo-generated current source in parallel with a diode, however a practical simulation requires adding series and shunt resistances to compensate the internal contacts and leakage currents. The most employed circuits to extract solar cell characteristics are the single diode model and the double diode model.

The single diode equivalent circuit is shown in Figure 5, where *D* is a parallel diode, R_{sh} is a shunt resistance that represents the leakage current, and R_s is a series resistance that serves as an internal resistance. The amount of electrical energy produced by the cell is represented by the current I_{ph}, which is proportional to the impinging radiation.

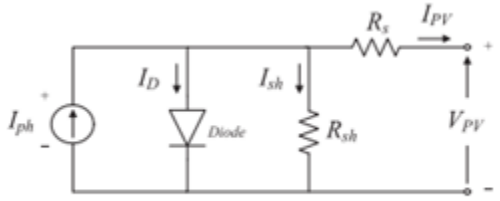

Figure 5:The single diode model of PV cells [6].

Using KCL, the mathematical equation that expresses the PV cell is given as follows:

$$I_{PV} = I_{ph} - I_D - \frac{V_D}{R_{sh}} \quad 1.1$$

Or

$$I_{PV} = I_{ph} - I_s \left[\exp\left(\frac{q(V_{PV} + R_s I_{PV})}{AKT}\right) - 1\right] - \frac{V_{PV} + R_s I_{PV}}{R_{sh}} \quad 1.2$$

Where:

$I_{ph}$is the light-generated current of the elementary solar cell.

I_d is the current of the parallel diode.

$I_s$is the reverse saturation current of the diode.

I_{PV} , V_{PV} are the output current and voltage of the PV cell respectively.

$q$is the electron charge (1.60217662 × 10^{-19} C).

A is the diode ideality factor and depends on the PV cell technology.

K is the Boltzmann constant (1.38064852 × $10^{-23} J/K°$)

T is the temperature of the p-n junction in unit kelvin.

The double diode equivalent circuit is more accurate, and it is formed by adding another parallel diode as depicted in figure 6:

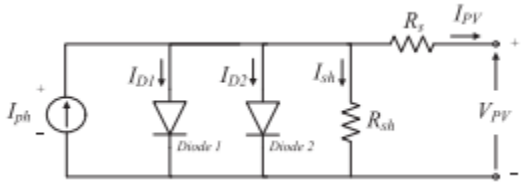

Figure 6:The Two-diode model of PV cell [6].

The output current-voltage characteristic of the double diode model is then:

$$I_{pv} = I_{ph} - I_{s1}\left[e^{\left(\frac{q(V_{pv}+R_S I_{pv})}{A_1 KT}\right)} - 1\right] - I_{s2}\left[e^{\left(\frac{q(V_{pv}+R_S I_{pv})}{A_2 KT}\right)} - 1\right] - \frac{V_{pv} + R_s I_{pv}}{R_{sh}} \quad 1.3$$

Usually, it is assumed that the PV modules that build a PV array have identical characteristics. If the connecting resistances among the PV modules are ignored, the single-diode-

model-based PV array's equivalent circuit is as shown in Figure 7. The relationship between the output voltage and current of the PV array in Figure 7 can be represented as [2]:

$$I_{pv} = N_P I_{ph} - N_P I_s \left[\exp\left(\frac{q\left(\frac{V_{pv}}{N_S} + \frac{R_S I_{pv}}{N_P}\right)}{AKT} \right) - 1 \right] - \frac{N_P}{R_{sh}} \left(\frac{V_{pv}}{N_S} + \frac{R_s I_{pv}}{N_P}\right) \quad 1.4$$

Where N_S is the number of series cells and N_P is the number of shunt cells.

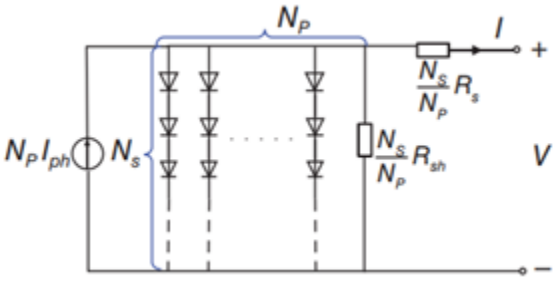

Figure 7: Single-diode model-based PV array's equivalent circuit.

1.6 PV Cell Characteristics:

Generally, the electric characteristics of PV cells are illustrated using current-versus-voltage curves and power-versus-voltage curves under different environmental conditions. Figure 8.a below, depicts the I-V and the P-V characteristics of an ideal PV cell exposed to different solar irradiation intensities, while figure 8.b depicts the cell characteristics under different temperature conditions. Three main points can be spotted from the two curves:

- **Output Short Circuit Current ($0, I_{SC}$)**

It is the maximum value of current that can be generated by a solar cell. Produced when the solar cell is short circuited**,** under this condition the short-circuit current and the light-generated current are almost identical.

- **Open Circuit Voltage ($V_{OC}, 0$)**

It is the maximum value of voltage a solar cell can provide at the output, and it is produced when no current is generated (I_{PV} =0). It reflects the voltage of the cell at night and can be expressed as [8]

$$V_{OC} = \frac{AKT}{q} \ln\left(\frac{I_{ph}}{I_S}\right) \quad 1.5$$

- **Maximum Power Point (MPP)**

It is the operating point at which the power delivered is maximum:

$$P_{MP} = V_{MPP} I_{MPP}$$

Using these parameters, it is possible to create a circuit model to simulate the cell performance.

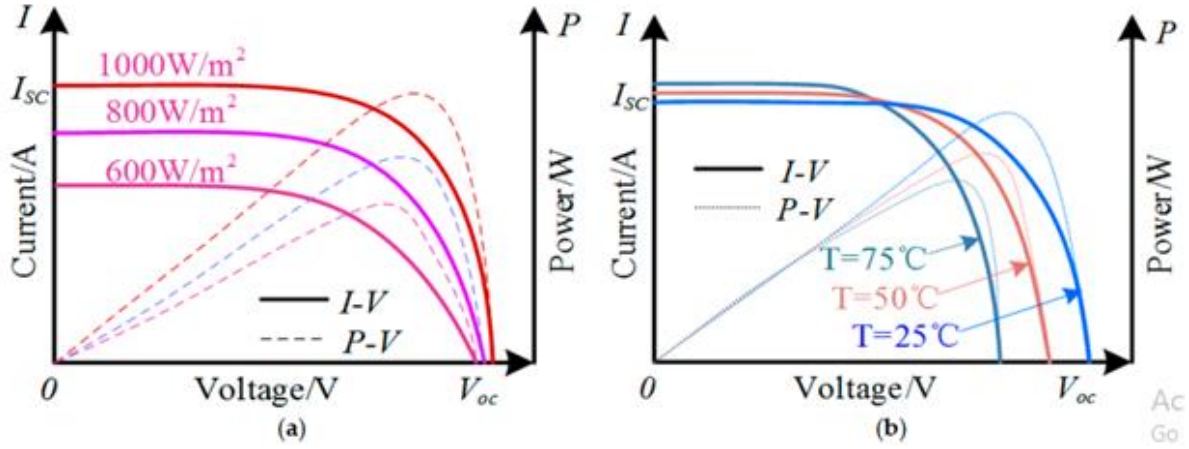

Figure 8:I-V and P-V characteristics of an ideal PV cell exposed to different: (a) solar irradiation intensities; (b)temperature conditions [7].

The two figures clearly show that the cell characteristics are highly non-linear and that the intensity of the solar radiation that hits the cell, controls the Short Circuit Current, while variations in the solar cell temperature significantly affect its Open Circuit Voltage (VOC).

It can be observed from Figure 8.a that the output current increases quasi-linearly with increasing solar irradiation, accordingly, the maximum power point of the module gets higher, while the output voltage increases very slightly.

On the other hand, Figure 8.b demonstrates that the PV voltage decreases with increasing temperature resulting in a net reduction in power, while has a small impact on the cell current.

The conversion efficiency and the Fill Factor are other key parameters that are used to evaluate the performance of solar cells, and compare between the different available PV technologies, to select the one that best suits the given operating conditions.

- **Power Conversion Efficiency (η)**

The conversion efficiency of a photovoltaic (PV) cell, is the percentage of the solar energy shining on a PV device that is converted into usable electricity [9]. It is defined as the ratio between the solar module output and incident light power:

$$\eta = \frac{P_{max}}{P_{in}} = \frac{V_{MPP} I_{MPP}}{A_C \times G} \quad (1.6)$$

where, $G$is the ambient radiation and A_C is the cell area.

- **Fill Factor *(FF)***

The fill factor is a measure of the real I-V characteristic. Its value is higher than 0.7 for good cells. The fill factor diminishes as the cell temperature is increased [10].

$$FF = \frac{P_{max}}{V_{OC} I_{SC}} \quad (1.7)$$

1.7 Partial Shading:

Under uniform insolation conditions, the P-V characteristic curve is unimodal and have only one peak. This tip represents the MPP of the solar panel, and it is the optimum operating point at which the power delivered to the load is maximum.

However, in most real life situations, the PV array is exposed to non-uniform insolation, in which some parts are shaded due to certain external conditions or surrounding objects (clouds, presence of trees, buildings), leading to potentially significant losses in the output power. In fact, shading in one single cell inside the PV modules can result in 90% of total power losses [11]. At that time, the current generated by the illuminated cells is greater than the current produced by the shaded cells; this mismatch makes the diode of the affected cells reverse biased; consequently, power will be wasted, which causes a hot spot problem and permanent damages to the PV panel [12]. These hot spots are sometimes invisible and lead to huge power dissipation in the form of heat, which may affect the neighboring cells. Thermal imaging cameras can be used for the inspection of solar panels and detection of these serious plights, before they lead to devastating impacts. To minimize the effects of partial shading and avoid deteriorations, a bypass diode is introduced in parallel to each PV module providing an alternative path for the current. Nonetheless, the insertion of a bypass diode will lead to the distortion of the IV and PV characteristics of the module and results in the occurrence of multiple peaks, among which there is only a single optimum power point denoted by GMPP (Global MPP), and it is the summit of the PV curve. This phenomenon is illustrated in figure 9.

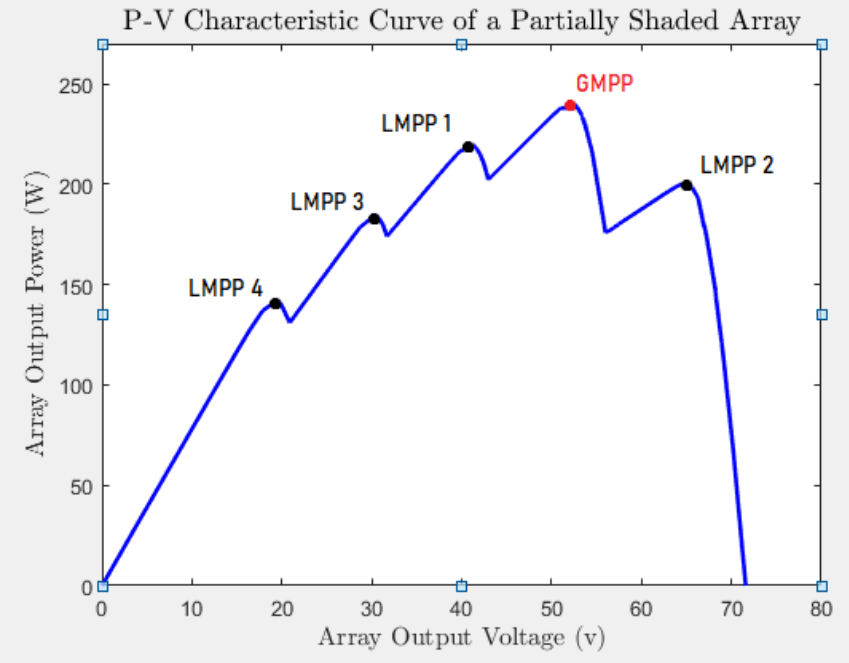

Figure 9:P-V Characteristic Curve of a Partially Shaded Array

1.8 DC-DC Converters:

1.8.1 Introduction:

DC-DC converters or Choppers, are electronic devices that are employed whenever a level conversion of DC voltage is required. In many ways, a DC-DC converter is the DC equivalent of a transformer [13]. Choppers are in general made up of capacitors, inductors and switches.

The key element in these devices is the controlled power switch driven by an external circuit. Usually, the switch is a semiconductor device that can be a: bipolar junction transistors (BJT), metal oxide semiconductor field effect transistor (MOSFET), gate turn off (GTO) thyristor, or an insulated gate bipolar transistor (IGBT).

The main concept in choppers, is that energy is periodically stored into and released from the magnetic field of the used inductor in other terms, by adjusting the duty cycle of the switching device, the amount of transferred power can be controlled.
Besides having two stages, DC-DC converters have two operating modes: continuous mode and discontinuous mode. In the continuous mode, the ON and OFF stages of the converter are in such a way that the current in the inductor never reaches zero. The discontinuous mode is just the opposite case.
There exist three main configurations of choppers:

- Buck Converter- Reduce the input voltage to a lower voltage level

- Boost Converter- Increase the input voltage to a higher level
- Buck-Boost Converter- Increase or reduce the input voltage

In this work we will focus on Boost converters, and throughout the subsequent analysis, the following assumptions will be taken:

1- The circuit is in steady state.
2- The components are ideal.
3- The capacitance is large enough such that the output voltage is held constant.
4- The switching period is T; the switch is ON for a time $t_{ON} = DT$ and off for time $t_{OFF} = (1 - D)T$
5- The converters are operating in continuous mode.

1.8.2 Step-up Converters:

Step up converters are used when the output voltage is required to be higher than the source voltage. The basic topology of a boost converter is shown in figure 10. For the sake of simplicity, the power controlled switch is replaced by a conventional one.

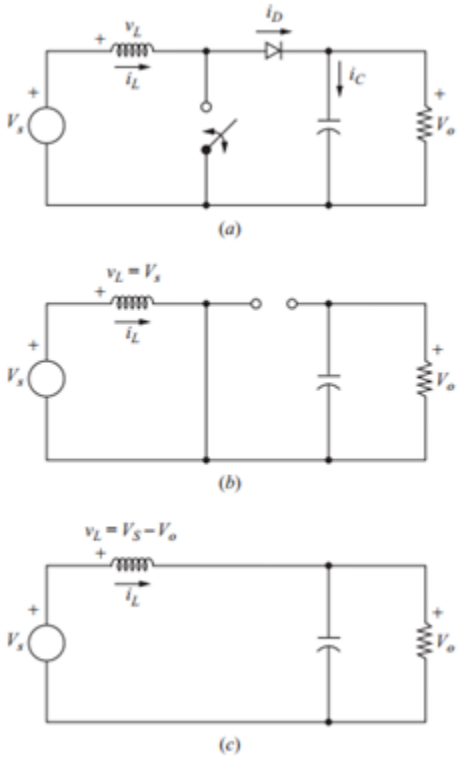

Figure 10: The Boost Converter; (a) Circuit; (b) Equivalent circuit for the switch closed; (c) Equivalent circuit for the switch open [14].

The basic operation of the step-up converter can be summarized as follows:

- When the switch turns on, the circuit of the boost converter reduces to the one of figure10.b, the current i_L flows from the source to the inductor, and energy builds up and stored.
- When the switch turns off, the circuit of the boost converter will be shown in figure 10.c. Since the inductor resists sudden variations in input current, its EMF is then reversed so that its voltage adds up to the input voltage, the stored energy will start to decay and the current flows through the inductor, the diode and the load, charging up the capacitor.

The resulting waveforms are shown in figure 11.

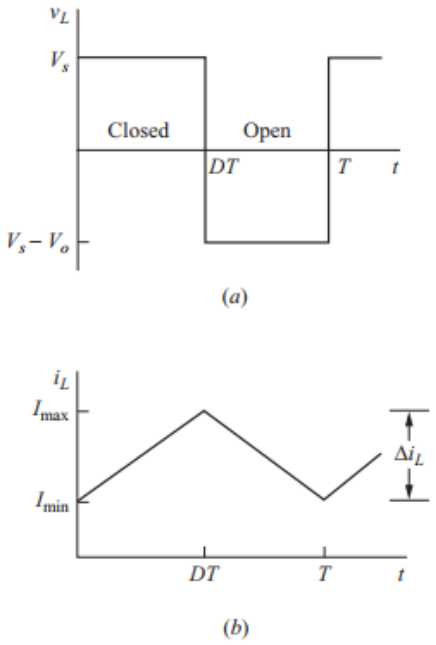

Figure 11:Boost Converter Output Waveforms; (a) Inductor voltage; (b) Inductor current [14].

Since the operation mode is in steady state, the average inductor voltage is zero:

$$\frac{1}{T}\int_0^T v_L dt = \left(\frac{1}{T}\int_0^{DT} v_L dt + \frac{1}{T}\int_{DT}^T v_L dt\right) = 0 \qquad 1.8$$

Following the waveform of figure 11.a, equation 1.8 is rewritten as follows:

$$DV_s + (V_s - V_0)(1 - D) = 0 \qquad 1.9$$

Which yields:

$$V_0 = \frac{V_s}{1 - D} \qquad 1.10$$

1.8.3 DC-DC Converters Design for PV systems:

As previously stated, DC-DC converters play a vital role in PV systems, a role that will be discussed in details in the next chapter. In this section we are going to provide a guideline in designing a proper boost converter circuit for PV systems.

1.8.3.1 Inductor Selection

During the T_{ON} state, the average inductor voltage is given as follows:

$$v_L = L\frac{di_L}{dt} \quad 1.11$$

Since the inductor current is changing linearly, then:

$$v_L = L\frac{\Delta i_L}{T_{ON}} = L\frac{\Delta i_L}{DT} \quad 1.12$$

Considering the fact that the average power supplied by the source, is equal to the average power absorbed by the load:

$$P_s = P_o \Rightarrow V_s I_L = \frac{{V_0}^2}{R} = \frac{{V_s}^2}{(1-D)^2 R} \quad 1.13$$

Hence, the average inductor current is given by:

$$I_L = \frac{V_s}{(1-D)^2 R} \quad 1.14$$

Looking at the inductor current waveform of figure 11.b, and based on the assumption that i_L is continuous (always positive), the following condition must be satisfied to guarantee the performance of the chopper in continuous conduction mode:

$$I_{min} = I_L - \frac{\Delta i_L}{2} > 0 \text{ which implies that } \frac{V_s}{(1-D)^2 R} - \frac{V_s DT}{2L} > 0 \quad 1.15$$

Therefore, the minimum inductance value can be obtained using equation 1.16:

$$L = \frac{D(1-D)^2 R}{2f} \quad 1.16$$

1.8.3.2 Selection of the output capacitor

The choice of C_o depends mainly on the peak-to-peak output ripple voltage which results from the fact that in practice, the capacitance has a finite value. During the ON state, the capacitor current can be expressed as:

$I_c = C_o\frac{\Delta V_o}{T_{on}}$ which implies: $C_o = \frac{DI_o}{\Delta V_o}$

The optimum output capacitance can then be calculated using

$$C_o \geq \frac{D}{R(\Delta V_o/V_o)f} \quad 1.17$$

Where f is the switching frequency ($f=1/T$)

In most cases, the desired output ripple voltage is set to be 2%, therefore:

$$C_o \geq \frac{D}{0.02Rf} \quad 1.18$$

1.8.3.3 Selection of the input capacitor

In many situations, an input capacitor is used to decrease the input voltage ripple and deliver an alternative current to the inductor [15]. The input capacitor current waveform is depicted in figure 12. Where the blue area serves as the charging phase in which $I_{C_{in}} \geq 0$.

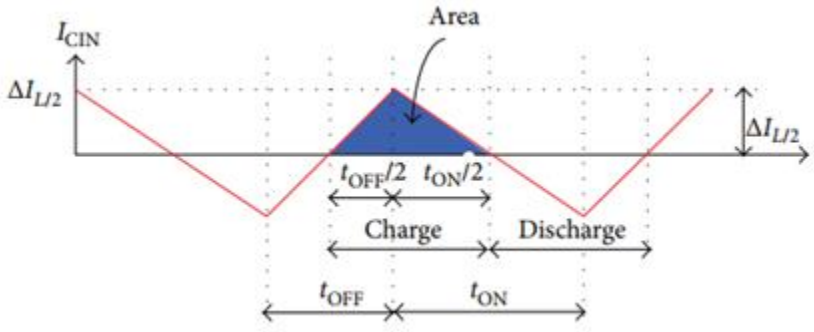

Figure 12:Current waveforms of the input capacitor in CCM [15].

The value of the desired input capacitance can then be calculated using equation 1.19:

$$C_o \geq \frac{D}{0.02Rf} \quad 1.19$$

Where the ripple voltage is set to be 2%.

1.9 Conclusion:

In this chapter, we have outlined the essentials of photovoltaic systems, their main types and key components. The working principle of solar cells has been set forth, along with their modeling, assessment parameters and characteristic curves, form which, the impact of ambient temperatures and solar irradiance levels were visualized and extracted. We have also straightened out the partial shading phenomenon and its harmful impacts on the solar system, we saw then how can PV panels

be protected using bypass diodes and how can this solution deform the IV and PV curves. The chapter then ended with a brief survey on DC-DC converters, laying out a guideline to design and properly select their components, to ensure an efficient and stable operation.

CHAPTER 2 : Maximum Power point tracking

2.1 Introduction:

We have seen that exogenous and weather conditions have a huge impact on the non-linear characteristics of PV cells. Fluctuations in sunlight levels, temperature, the connected load, as well as the surrounding objects, orientation of the panel and even dust are few factors that may lead to a substantial change in the power extracted from the PV system, consequently the operating point will be jiggling accordingly throughout the day. Moreover, protecting the system using bypass diodes from the harmful effects of partial shading phenomenon, will lead to multimodal characteristics in the P-V curves of the solar array. For these reasons, maximum power point tracking techniques have gained an extensive attention in literature to come up with powerful solutions and extenuate these sticky issues.

MPPT controllers are electronic devices usually integrated with dc-dc converters that allow locating the point at which the PV system operates in its optimum state, and hence draw the maximum possible power out from PV panels, regardless of the operating conditions. In literature, a myriad number of MPPT techniques have been proposed, each of which has its own advantages and disadvantages, and differ in many aspects. This chapter steps through a brief introduction into MPPT theory and some of the most popular classical algorithms applied in PV systems.

2.2 Load Matching:

When the solar panel is directly coupled to the load, the operating point will be at the intersection of its I-V curve and the load line that has a slope of $1/R_{load}$. Usually, this intersection point is not located at the panel MPP, and a significant amount of power will be wasted. An intermediate driver has to be connected between the PV generator and the load side of the system, which allows adjusting the impedance seen by the PV array to match its MPP level. To fulfill this role, a power interface that consists of an MPPT controller linked to a DC-DC converter, is placed between the PVG and the other side of the system. The duty cycle of the power converter will then be properly adjusted whenever a change in atmospheric conditions is detected, and prospect the optimum state of the system under the given circumstances.

A typical MPPT based PV system is shown in figure 13. The solar panels generate electric energy according to the amount of sunlight falling on their surface. The generated power is transferred into a DC-DC converter, driven by an MPPT controller that receives the real time operating parameters, specifically: the panel voltage and current. The MPP tracker then based on the acquired information and the built-in algorithm, works out the required adjustments in the duty cycle of the power converter, in order to locate the MPP associated with the operating conditions, and maintain the system in its optimum state.

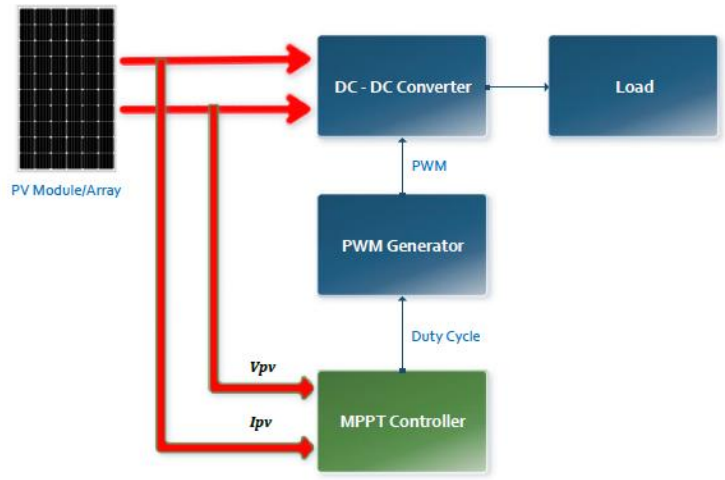

Figure 13:Typical MPPT based PV system.

The duty cycle controls the load resistance seen by the panel, and hence the mismatch between the two sides will be nulled. Consider a boost converter used as an adapter between the source and the load [15]:

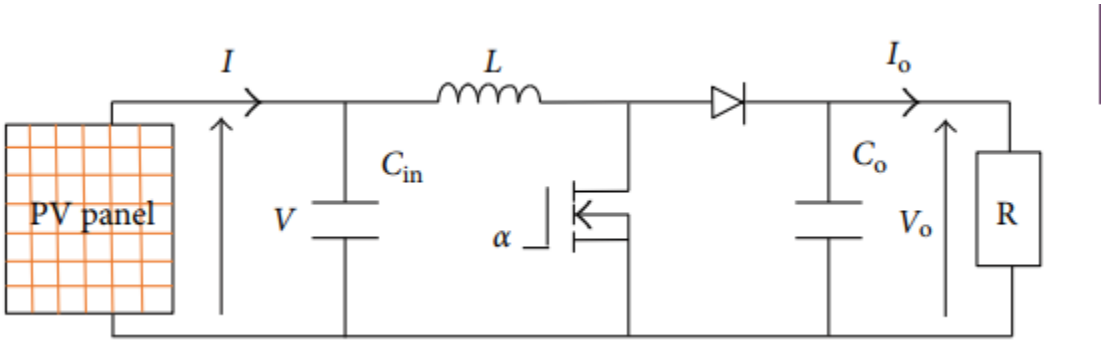

Figure 14: Boost Converter in an MPPT based PV System

From section 1.8, the average out sides voltage and current are given by:

$$V_o = \frac{V_{in}}{1 - D} \tag{2.1}$$

$$I_o = (1 - D)I_{in} \tag{2.2}$$

Using equations 2.1 and 2.2, the relationship between the input resistance seen by the PV module (R_{equiv}) and the output load impedance can be extracted as follows:

$$R_{equiv} = \frac{V_{in}}{I_{in}} = \frac{(1-D)V_o}{\frac{I_o}{1-D}} = (1-D)\frac{V_o}{I_o} = (1-D)R_{load} \quad 2.3$$

The duty cycle required for the MPPT controller to locate and extract the maximum power can then be found using equation 2.4:

$$D = 1 - \sqrt{\frac{R_{equiv}}{R_{load}}} \quad 2.4$$

Where R_{load} is the resistance at which power is at its optimal point:

$$R_{load} = \frac{V_{MPP}}{I_{MPP}} \quad 2.5$$

2.3 Classical MPPT techniques:

The most popular classical strategies employed to track the MPP are inspired from the hill-climbing principle, which are well known for their ease of implementation and good performance when the irradiation is nearly constant. The idea consists of moving the operating point of the PV array in the direction in which the power increases [5]. In this section, we are going to provide a general overview of the two most widely employed conventional MPPT techniques and their merits and limitations.

2.3.1 Perturb and observe (P&O) technique:

P&O is one of the most commonly known technique in tracking the MPP. The simplicity and ease of implementation makes it an attractive approach to use since it only requires a single voltage sensor to measure the PV panel voltage, and a single current sensor to measure PV current.

- **Principle of operation:**

The algorithm keeps periodically injecting a perturbation on the duty cycle of the converter and hence a perturbation in the operating voltage of the DC-link between the PV array and the power converter [18]. The output PV power is then recorded and compared to its antecedent value. If the power has been increased, the sign of the perturbation is kept the same until the MPP is reached, if at a certain step, the power is decreased, the direction of the perturbation should be reversed. The operating point then remains oscillating around the peak power point. The procedure of the P&O technique is summarized below and illustrated in the flowchart of figure 15:

Step 1 Measure the PV current and voltage.

Step 2 Calculate the corresponding Power.

Step 3 Repeat steps 1 and 2 with new voltage and current measurements.

Step 4 Calculate the variation in power and voltage.

Step 5 The direction of climbing to the MPP will be decided according to the results of step 4, and the voltage will be increased or decreased through proper adjustments of the duty cycle based on the fact that the duty cycle is inversely proportional to voltage:

- When both power and voltage are increased (variation is positive), the OP is on the left side of the P-V curve and the voltage should be incremented (decrease duty cycle).
- When the power variation is positive and the voltage variation is negative, the OP is on the right-hand side and the PV voltage is to be decremented **(**increase duty cycle)**.**
- When the power variation is negative and the voltage variation is positive, the OP is on the right side of the P-V curve and the voltage should be decremented.
- When both power and voltage variations are smaller than zero, the OP is on the left side and the PV voltage should be incremented.

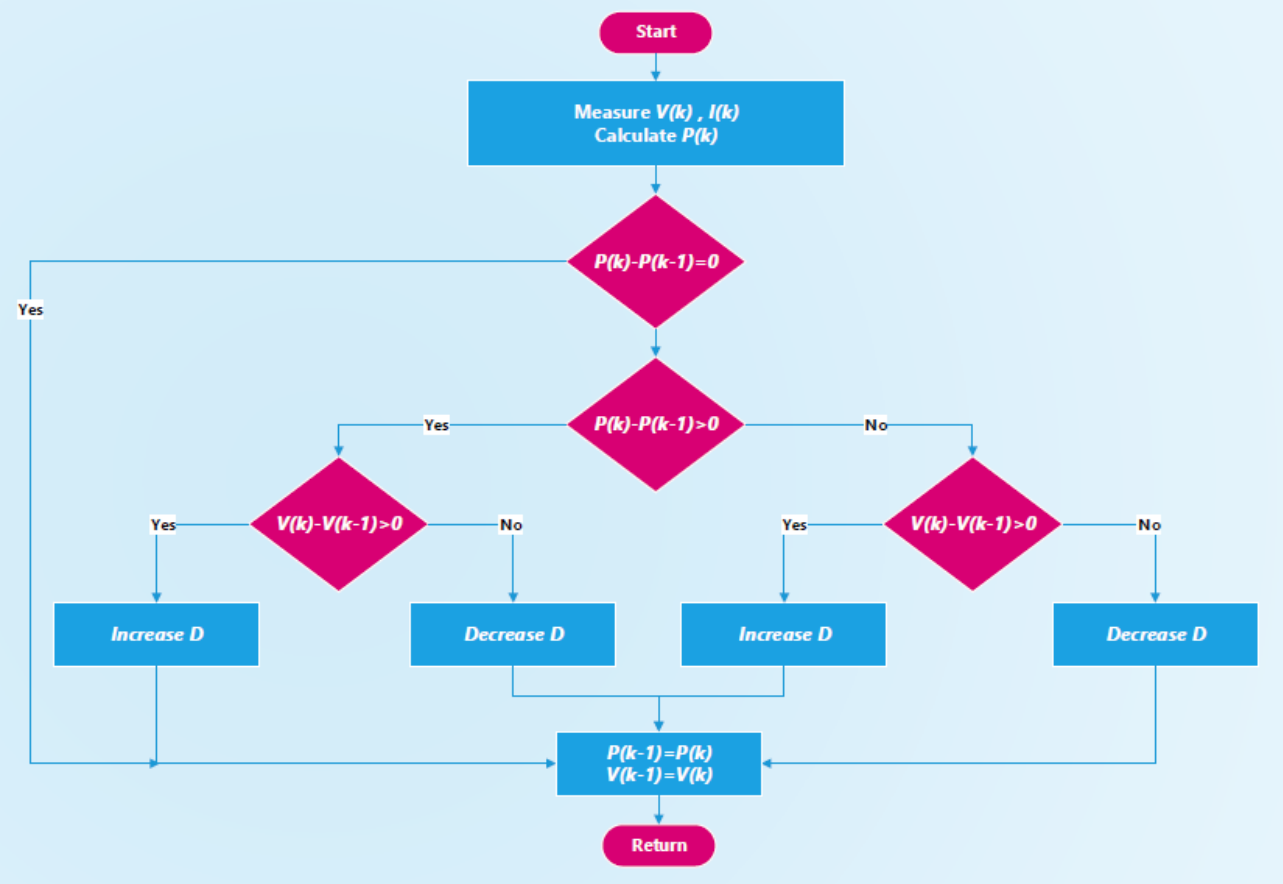

Figure 15: P&O based MPPT.

2.3.2 Incremental conductance (IC) technique:

The IC approach is derived from the fact that the slope of the PV panel characteristics at the MPP in the P-V curve is zero, that is:

$$\frac{\partial P_{PV}}{\partial V_{PV}} = V_{PV} \times \frac{\partial I_{PV}}{\partial V_{PV}} + I_{PV} = 0 \qquad 2.6$$

Equation 2.6 can be reformed as follows

$$\frac{\partial I_{PV}}{\partial V_{PV}} = -\frac{I_{PV}}{V_{PV}} \approx \frac{\Delta I}{\Delta V} \qquad 2.7$$

Where ΔI , ΔV are the increments in the PV panel current and voltage respectively.

The governing equations of the IC algorithm are then derived from the P-V characteristics and can be modeled as follows:

$$\begin{cases} \frac{dP}{dV} < 0 \text{ if } \frac{I_{PV}}{V_{PV}} < -\frac{\partial I_{PV}}{\partial V_{PV}} & \text{on the right of MPP} \\ \frac{dP}{dV} = 0 \text{ if } \frac{I_{PV}}{V_{PV}} = -\frac{\partial I_{PV}}{\partial V_{PV}} & \text{at the MPP} \\ \frac{dP}{dV} > 0 \text{ if } \frac{I_{PV}}{V_{PV}} > -\frac{\partial I_{PV}}{\partial V_{PV}} & \text{on left of MPP} \end{cases} \qquad 2.8$$

Which are used to find out the location of the operation point and select the direction of the perturbation. It can be seen that the left-hand side of the above inequations constraints represents the incremental conductance of the P-V panel whereas the right-hand side denotes the instantaneous conductance.

- **Principle of operation:**

The IC algorithm starts the cycle by sensing the current and voltage values, and then comparing the instantaneous conductance **(I/V)** to the incremental conductance **(**ΔI/ΔV**)** based on the previous stated rules. The flowchart illustrating the technique is shown in figure 16.

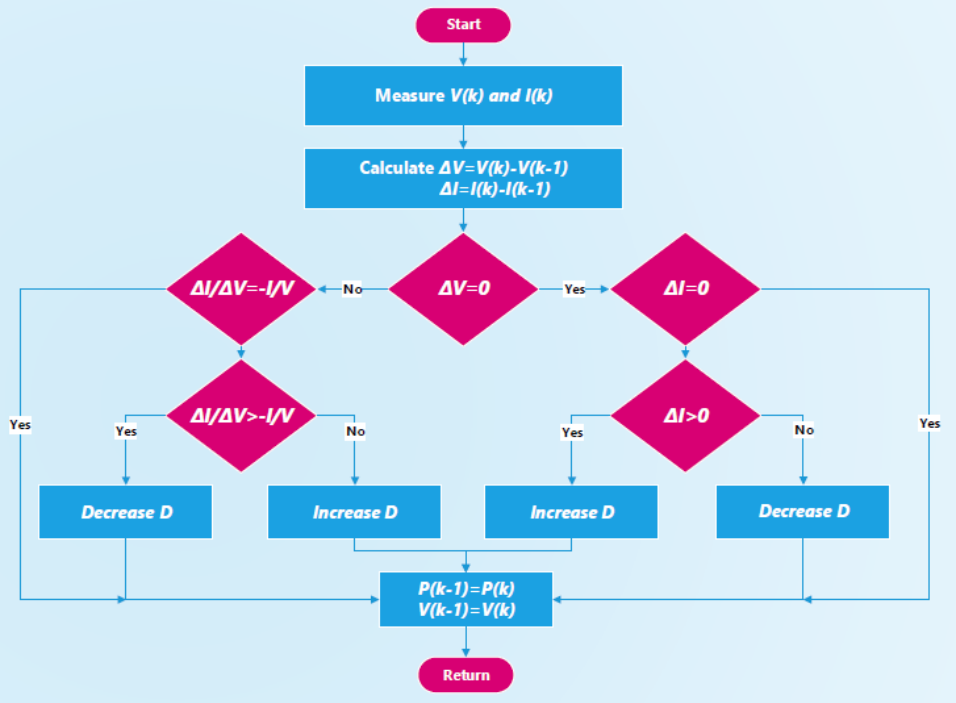

Figure 16: IC based MPPT.

2.4 Simulation and results:

2.4.1 Overall system:

To evaluate the two techniques, a standalone PV system that consists of two serially connected PV arrays, each of which is composed of two parallel strings, along with a boost converter, is designed using Simulink and Matlab. The overall system is illustrated in figure 17.

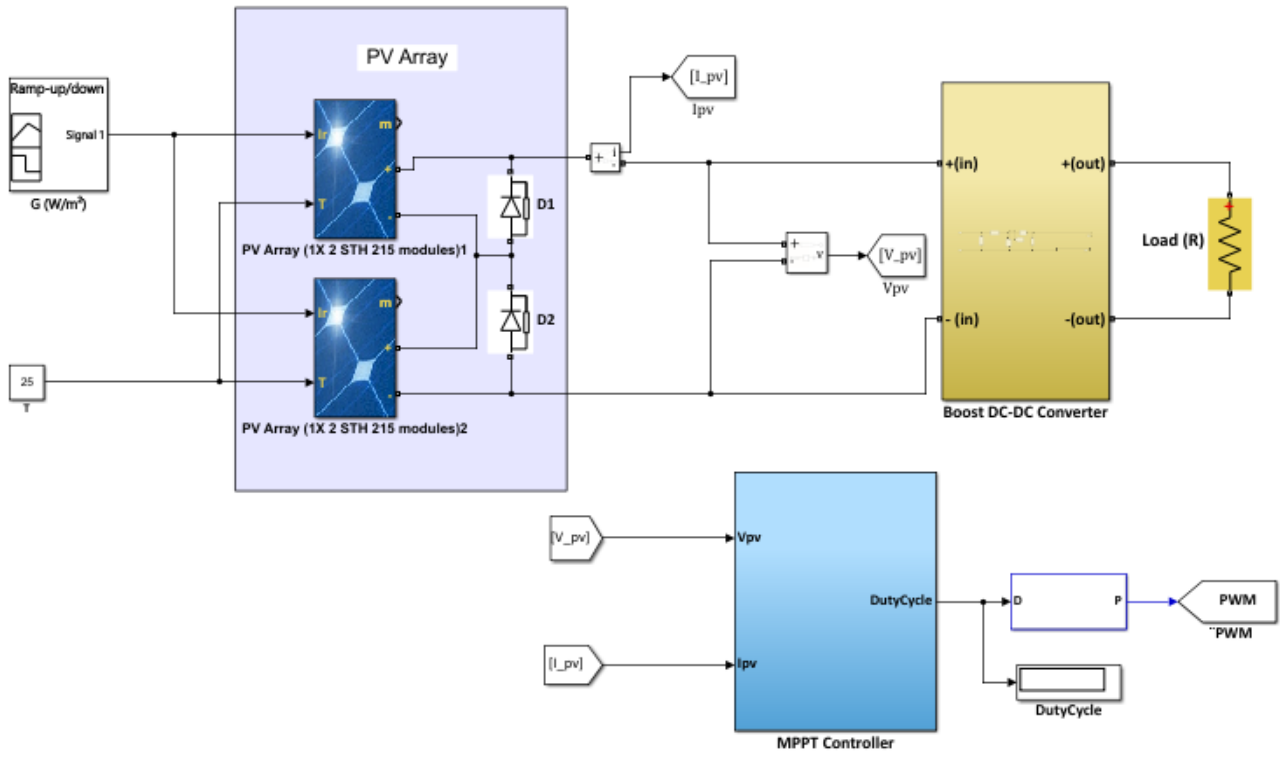

Figure 17: Simulink Model of the Designed System -1-

2.4.2 PV array Configuration:

The photovoltaic panel model used throughout this work is 1Soltech 1STH-215-P and has the following characteristics:

Maximum Power (W)	Cells per module (Ncell)	Diode saturation current I0 (A)
213.15	60	2.9259e-10
Open circuit voltage Voc (V)	Short-circuit current Isc (A)	Diode ideality factor
36.3	7.84	0.98117
Voltage at maximum power point Vmp (V)	Current at maximum power point Imp (A)	Shunt resistance Rsh (ohms)
29	7.35	313.3991
Temperature coefficient of Voc (%/deg.C)	Temperature coefficient of Isc (%/deg.C)	Series resistance Rs (ohms)
-0.36099	0.102	0.39383

Figure 18: Characteristics of the 1Soltech 1STH-215-P PV Panel.

The P-V characteristics of the used panel under three levels of uniform solar irradiation are depicted in figure 19.

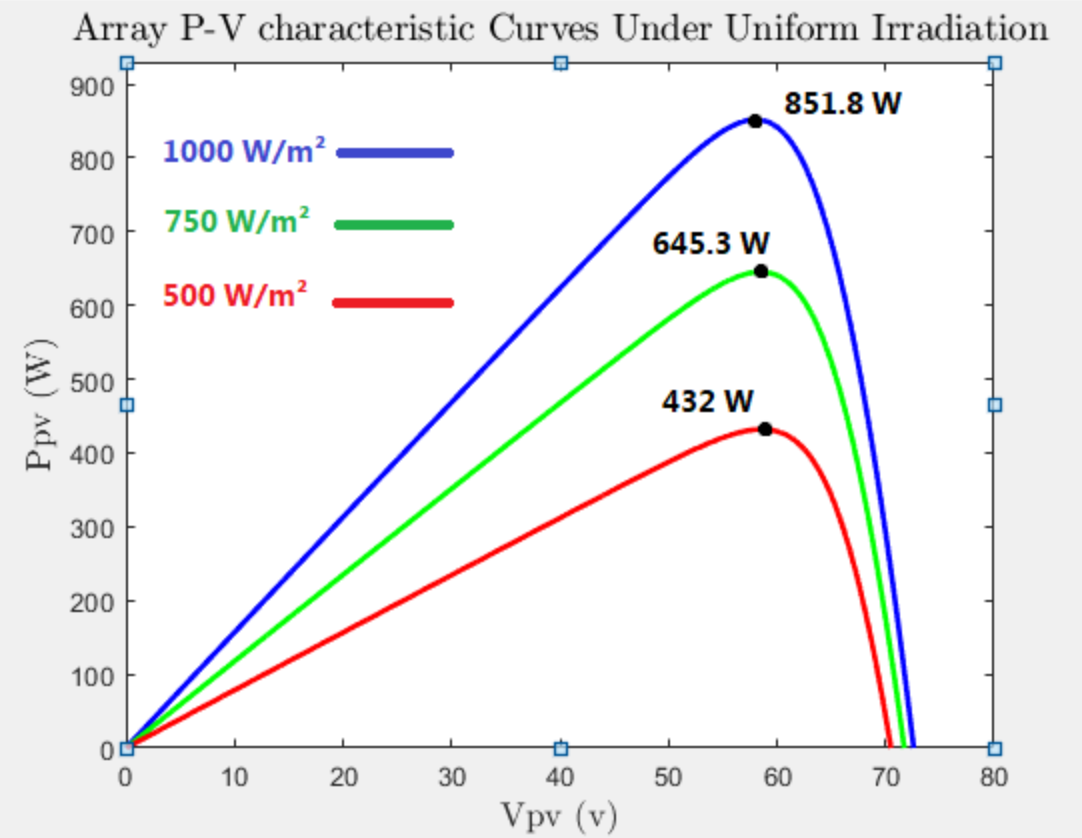

Figure 19: The P-V characteristics of the panel under three levels of uniform solar irradiation.

2.4.3 DC-DC boost converter design:

Setting the switching frequency to be 5 kHz and using the results of section 1.8, the components of the power converter are provided in table 1.

Table 1: DC-DC Converter Components Selection for System 1

Switching Frequency	**Inductor (L)**	**Input Capacitor**	**Output Capacitor**
5 kHz	2 mH	100μF	100μF

2.4.4 Simulation:

2.4.4.1 Uniform irradiation:

The PV panel is first exposed to a fast-varying uniform solar irradiation of three intensity levels (500, 750 and 1000/m²), each of which lasts for 0.25 seconds. The resulting power, current and voltage curves of each method are shown in figures 20 and 21.

a- P&O

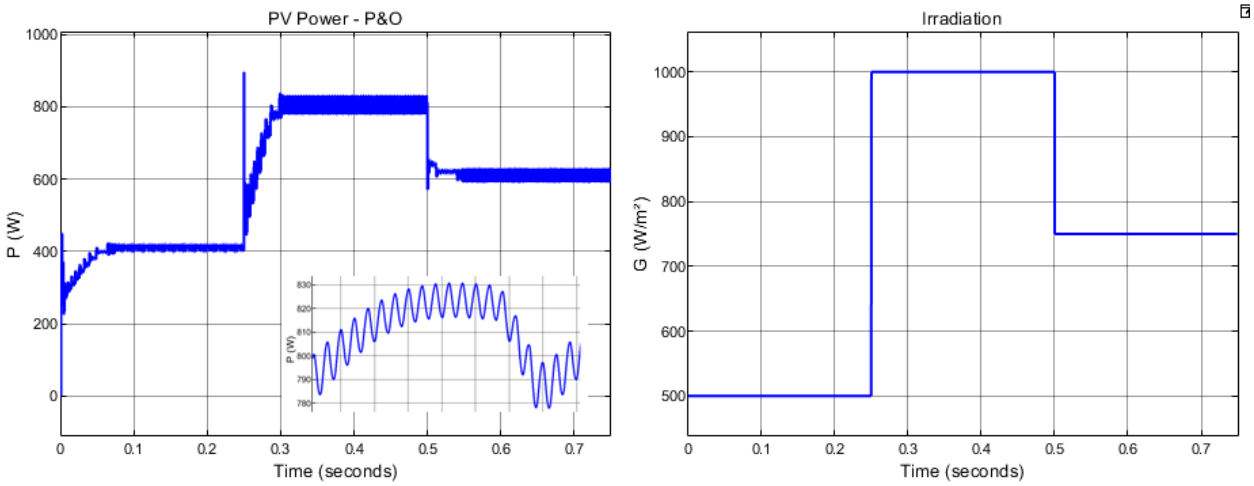

Figure 20: Resulting curves under fast varying uniform irradiation using the P&O technique.

b- IC

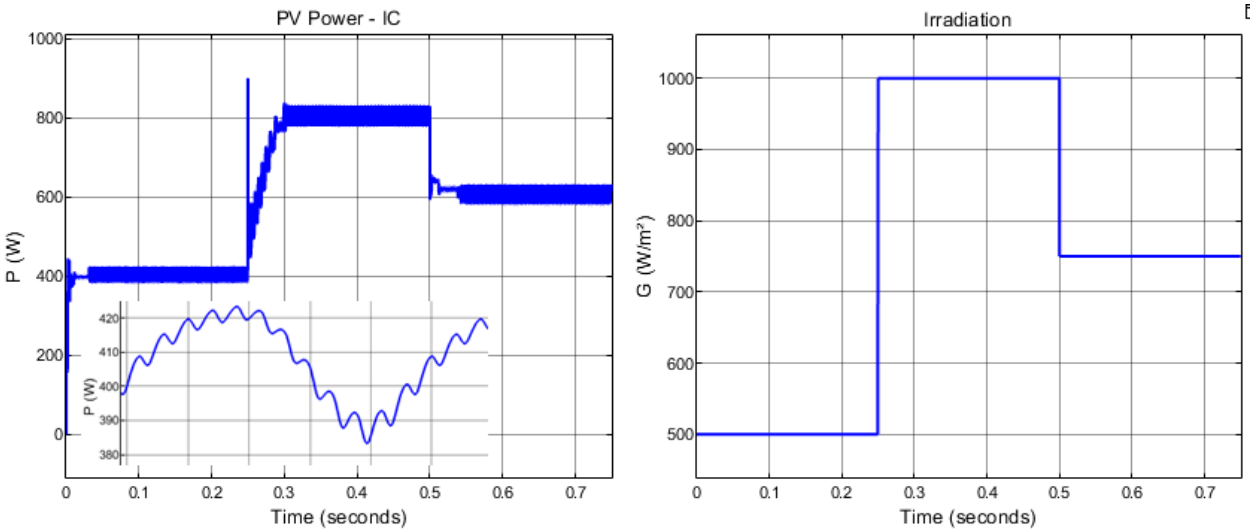

Figure 21: Resulting curves under fast varying uniform irradiation using the IC technique.

2.4.4.2 Non-uniform Conditions :

The system will now undergo partial shading as depicted in figure 22.

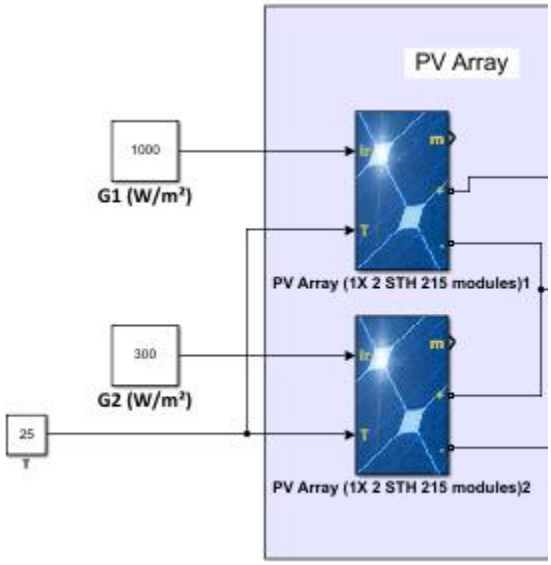

Figure 22: PV Panel Subjected to Non-Uniform Irradiation

The resulting P-V curve consists of three peaks as shown in figure 23.

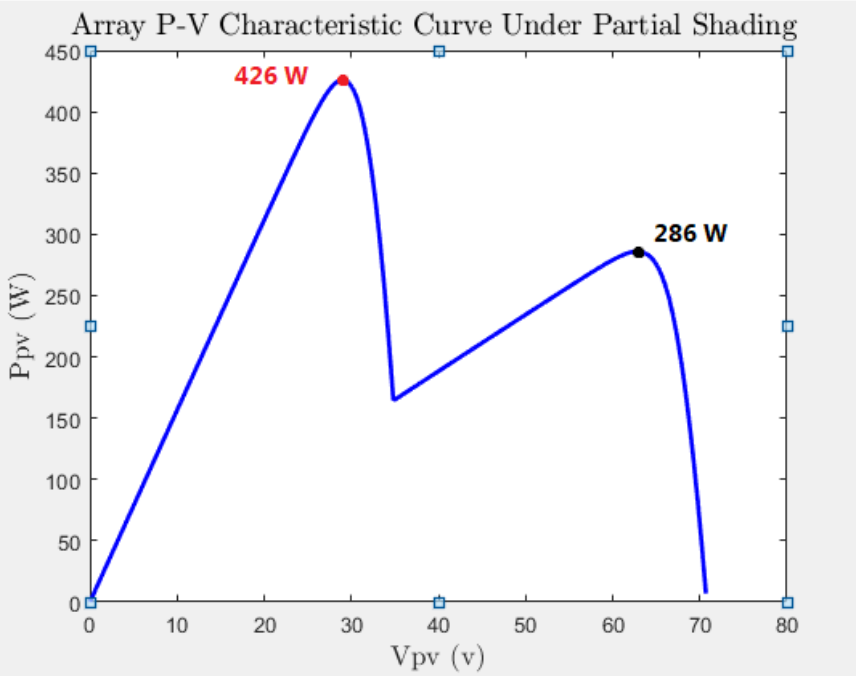

Figure 23: Module P-V characteristic curve under partial shading.

a- P&O

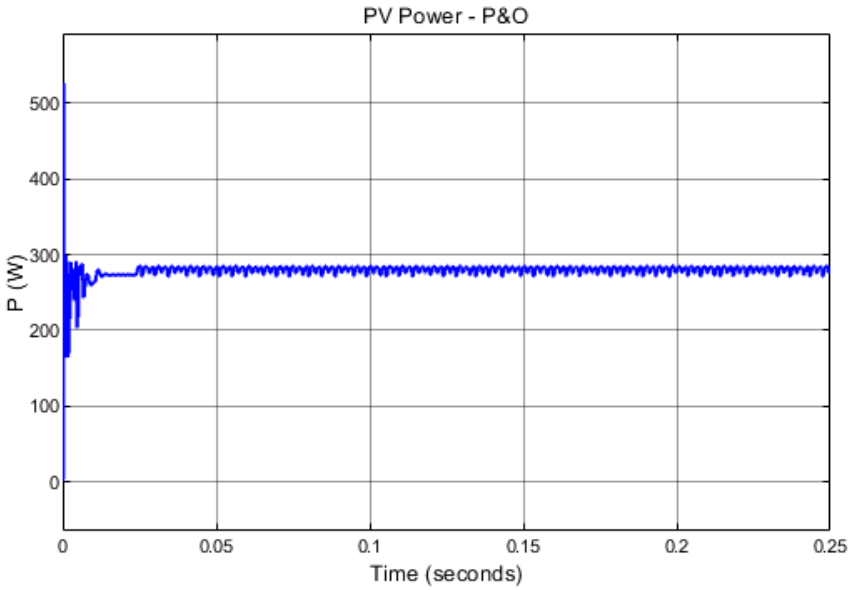

Figure 24:Resulting Curves Under Non-Uniform Irradiation Using the P&O Technique.

b- IC

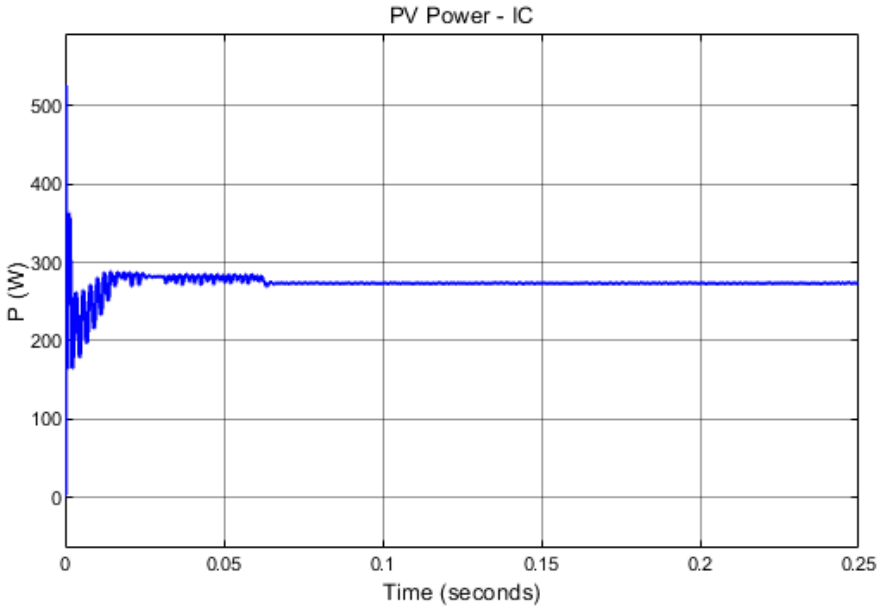

Figure 25:Resulting Curves Under Non-Uniform Irradiation Using the IC Technique.

2.5 Results and discussion:

The simulation results demonstrated two major drawbacks:

- The first drawback is the appearance of fluctuations around the MPP, this is due to the fixed step size taken at each sampling instant, which forces the operating point to keep going back and forth around the MPP without staying still. Controlling the amplitude of these oscillations leads to a fundamental tradeoff between tracking accuracy and tracking time. Enhancing the steady state requires a small size of increments the reference voltage will be taking, on the other hand, decreasing the step size will increase convergence duration.

- The second drawback is their big failure to distinguish between the global optimum point and the remaining peaks in partial shading conditions. The two techniques got trapped around the LMPP, which makes them unsuitable in most real life situations.

CHAPTER 3 : Soft computing algorithms and Novel Techniques

3.1 Introduction:

Because the afore discussed classical techniques have demonstrated several shortcomings in tracking the MPP, particularly, their inability to handle partial shading conditions, numerous soft computing and artificial intelligence algorithms have been proposed to cope with such situations and improve the dynamic and steady-state tracking performance. Most of these alternatives employ metaheuristics which had shown great performance regardless of the operating conditions.

Nature-inspired algorithms, are a family of metaheuristic optimizers that are based on imitating certain biological, social processes and physical laws commonly found in nature. These methods have attracted intensive research interest due to their efficiency in solving highly non-linear optimization problems with real-world engineering applications.

Almost all metaheuristics consist of two essential strategies: intensification and diversification. The Diversification phase is devoted to generate disperse solutions and efficiently scout the search space via randomization. In this way, the algorithm will have high prospects to find promising regions where the global optimal solution might be located, and hence avoids getting trapped in local peaks. In the other hand, intensification intends to inspect the local neighborhood around the best solutions and improve their quality. Intensification and diversification are commonly termed exploitation and exploration respectively.

The most prominent nature-inspired algorithms are the genetic algorithms, particle swarm optimization, Ant Colony Optimization, Grey wolf optimization, Whale Optimization Algorithm and so many others. The reader is referred to [17,18] to gain some fundamental insights in the field, and a comprehensive overview about different nature-inspired techniques and their most popular applications.

In this chapter, we will step through several metaheuristic algorithms, among which, three recent techniques along with their MPPT implementations are laid out.

3.2 Particle Swarm Optimization:

3.2.1 Inspiration:

Particle swarm optimization (PSO) is a population based stochastic optimization technique initially developed by Dr. Eberhart and Dr. Kennedy in 1995, and it is one the most widely used algorithms in many optimization engineering problems. It is inspired by the social behavior and swarm intelligence of animals in nature, namely bird flocking and fish schooling.

3.2.2 Mathematical Modelling and Process Steps:

The particle swarm optimization uses a number of particles that form a swarm, wandering around the search space looking for the best solution. Initially, the size of the swarm is defined, and each particle is assigned a random position within the search space to be a candidate solution. The "goodness" of each individual is evaluated by the fitness function related to the optimization problem. In each iteration, the position of each particle is updated based on its personal best location visited so far denoted P_{best} and the position of the most successful particle in the whole population denoted by G_{best} (Global best solution).

At iteration t+1, the population is updated using the following two equations:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad 3.2.1$$

$$v_i^{t+1} = \omega v_i^t + C_1 r_1 (P_{best}, i - x_i^t) + C_2 r_2 (G_{best} - x_i^t) \quad 3.2.2$$

$$i = 1, 2, \dots\dots\dots, N$$

Where v_i^t and $x_i^t$are respectively the velocity and position of the ith particlewithin t iterations, $\omega$is the inertia weight, C_1 and C_2 are the acceleration coefficients, r_1 and r_2 are two generated random numbers that are uniformly distributed in the interval [0, 1].P_{best}, i is the personal best position of particle i achieved so far ,and $G_{best}$is the global best position**.** Figure 26 depicts the movement of particles in the optimization process.

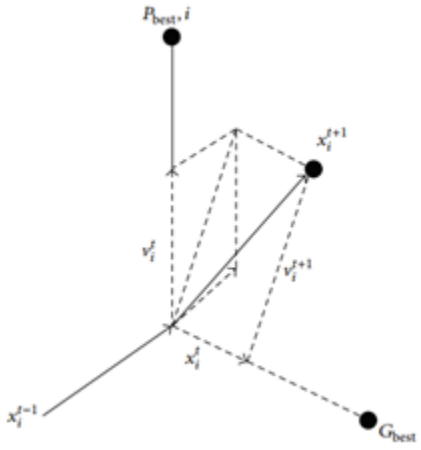

Figure 26: Illustration of a Particle's movement during the optimization process [19].

The process of the PSO algorithm can be summarized as follows:

Step1: Initialize the number of particles N_P, maximum number of iterations T, ω, C_1 and C_2

Step 2: Initialize a population of particles, and randomly set the position and velocity vectors of each particle.

Step 3: Set iteration t=1.

Step 4: Evaluate the fitness of each particle.

Step 5: Set $P_{best}, i = x_i^t$, and G_{best} to be the particle having the best position.

Step 6: Update the individuals' velocities and positions using equations 3.2.1- 3.2.2

Step 7: Evaluate the fitness of the updated population from step 6, and find the particle having the best position (x_{best}^{t+1}).

Step 8: If the fitness value of the updated ith particle is better than its antecedent fitness, update

$P_{best}, i = x_i^{t+1}$

Step 9: If the fitness value of the best particle at the current iteration obtained from step 7 is better than the fitness of G_{best}, update $G_{best} = x_{best}^{t+1}$

Step 10: If $t < T$ or the convergence criterion is not fulfilled, then update t=t+1 and repeat steps from 6 to 10, else go to step 11

Step 11: Return the Global best solution G_{best}.

3.2.3 PSO based MPPT:

In MPPT applications, the objective function is to maximize the output PV power as follows:

$$P(d_i^{t+1}) > P(d_i^t)$$

Where $P(d_i^k)$ is the power associated with the ith transmitted duty cycle at the tth iteration.

The PSO technique can be appointed to perform the task of maximum power tracking by initially defining a set of N_P several duty cycle solutions to act as the positions of particles. The search process will then be carried out as given in the process steps stated above, where G_{best} represents the duty cycle that produces the largest PV output power. The procedure of tracking the peak power point can be summarized as follows:

Step 1: Initialize the number of duty cycle solutionsN_P, maximum number of iterations$T$, ω, $C_1$and C_2

Step 2: A solution vector of N_P particles representing a set of duty cycle solutions and a velocity vector of the same length are initialized.

Step 3: The generated duty cycles are transmitted to the boost converter and the corresponding PV power is evaluated.

Step 4: Set $P_{best}, i = d_i$ in the first iteration and obtain G_{best}.

Step 5: Update the duty cycles using equations 3.2.1 and 3.2.2.

Step 6: The updated duty cycles are transmitted to the power converter and the output PV power of each is evaluated.

Step 7: Update P_{best}, i and G_{best} if better solutions have been obtained.

Step 8: If $t < T$ or the convergence criterion is not fulfilled, iteration is incremented and steps 5 through 7 are repeated. Else, go to step 9

Step 9: Transmit the global best duty cycle G_{best} to the power converter.

The algorithm has to be reinitialized whenever a change in weather conditions is detected, according to the following inequality constraint:

$$\frac{\left|P_{pv-new} - P_{pv-previous}\right|}{P_{pv-previous}} \geq C \qquad 3.2.3$$

Where C can be set to 0.1.

The flowchart illustrating PSO based MPPT is depicted in figure 27.

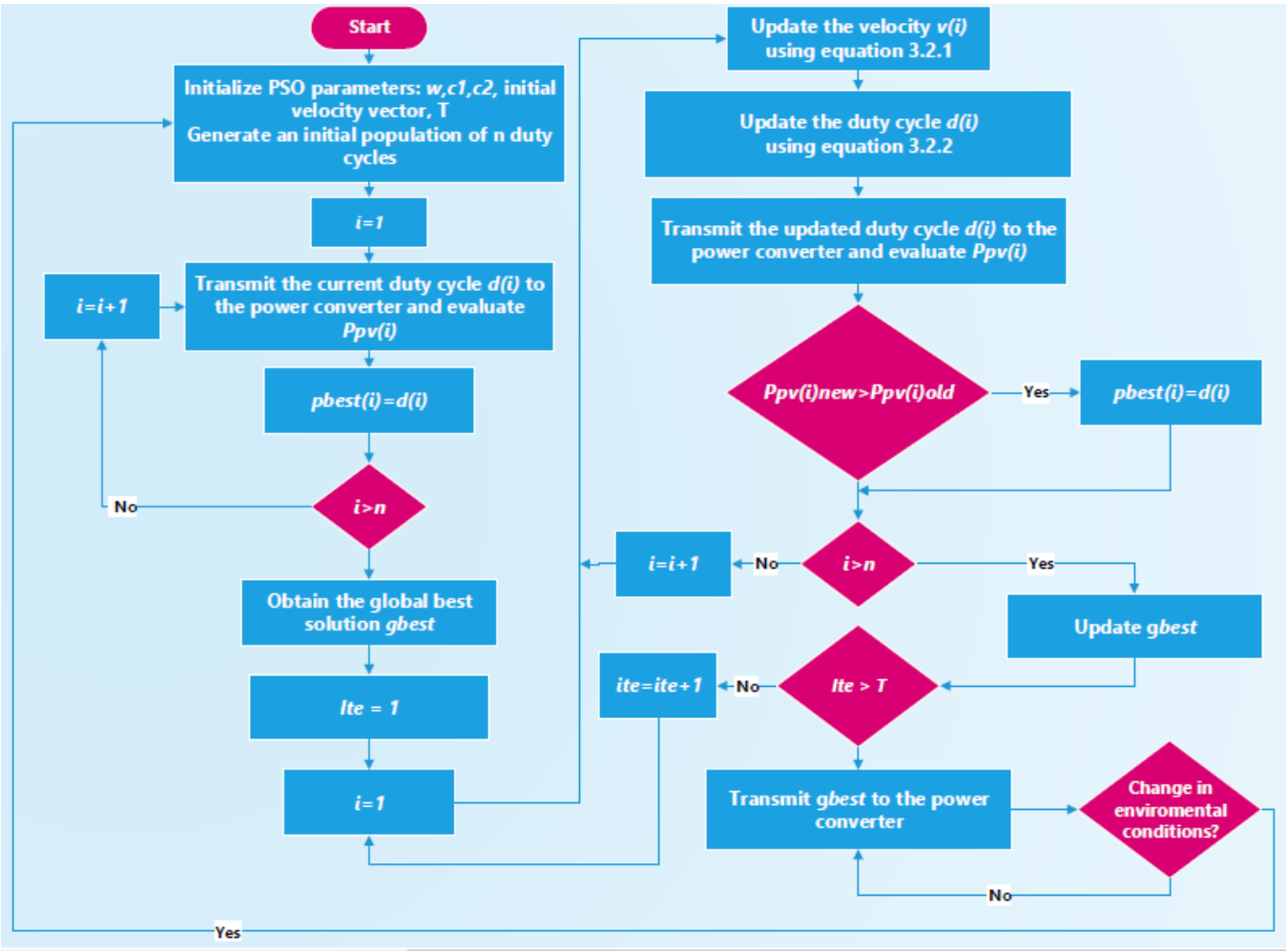

Figure 27: PSO based MPPT.

3.3 Whale Optimization Algorithm:

3.3.1 Inspiration:

The Whale optimization Algorithm is a nature-inspired metaheuristic population based technique proposed by Seyedali Mirjalili and Andrew Lewis in 2016 [20]. It is inspired from the fascinating hunting strategy used by a family of whales, known as *bubble-net feeding*. This behavior is employed by Humpback whales to trap small fishes, by floating around in a circular pattern, and blowing bubbles to surround the scooped fishes as depicted in figure 28.

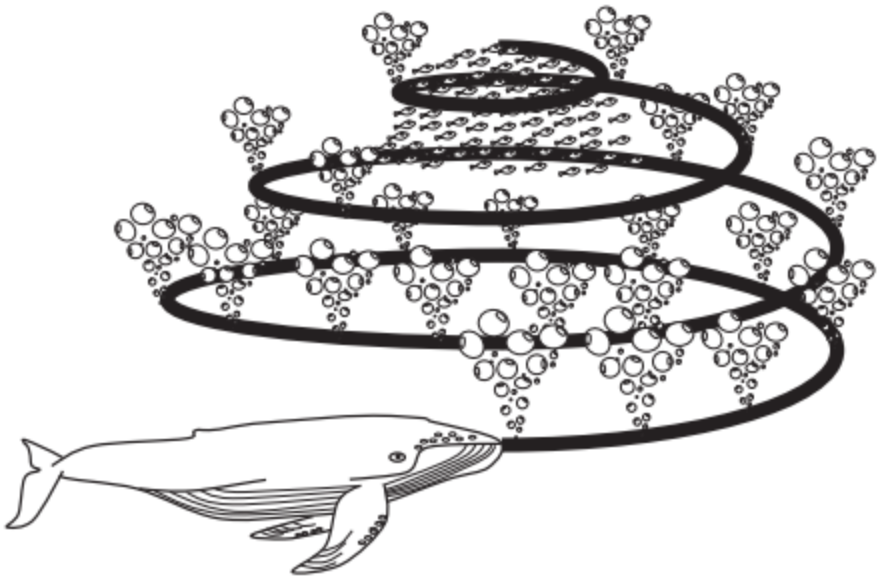

Figure 28 : Bubble-Net Feeding Behavior of Humpback Whales.

3.3.2 Mathematical Modeling:

The WOA is composed of three main strategies:

A- Search for prey:

In this phase, the whales will explore the search space looking for preys. Their positions are updated randomly and moved away from each other to perform a global search, using the following equations:

$$\vec{X}(t+1) = \overrightarrow{X_{rand}}(t) - \vec{A} \cdot \vec{D} \quad 3.3.1$$

$$\vec{D} = \left|\vec{C} \otimes \overrightarrow{X_{rand}}(t) - \vec{X}(t)\right| \quad 3.3.2$$

$$\vec{A} = 2a\vec{r} - a \quad 3.3.3$$

$$\vec{C} = 2\vec{r} \quad 3.3.4$$

$$a = 2(t - \frac{t}{T}) \quad 3.3.5$$

Where t represents the current iteration, $\overrightarrow{X_{rand}}(t)$ is a random position vector chosen from the current population. $\vec{A}$ *and* $\vec{C}$ are coefficient vectors and a is linearly decreased within the interval [2,0] over the course of iterations.

This mode of operation which represents the exploration phase is performed when $\left|\vec{A}\right| > 1$.

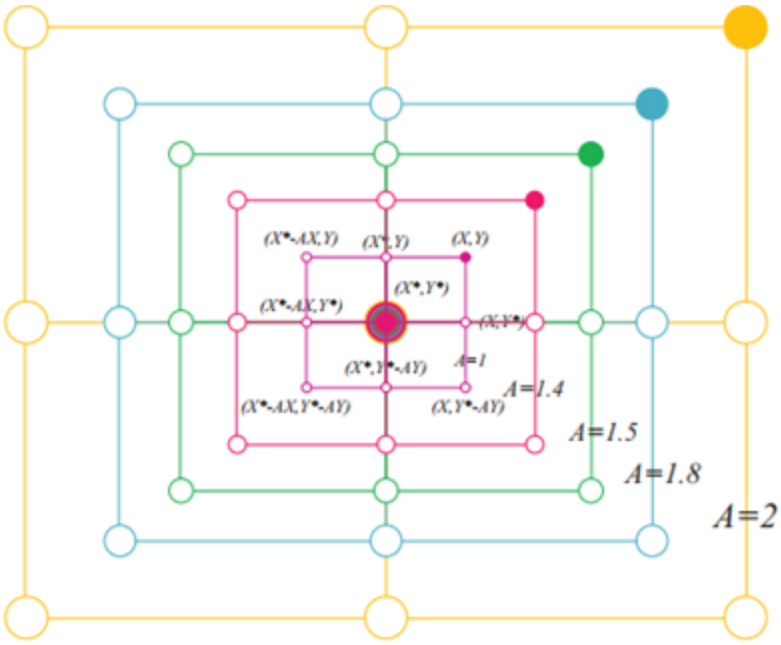

Figure 29Exploration mechanism implemented in WOA (X is a randomly chosen search agent) [20].*

B- Bubble net hunting:

The modeling of this behavior is carried out using two strategies:

- *Shrinking Encircling mechanism:*

Looking at equations 3.3.4 and 3.3.5, it can be observed that the value of A is always within the range $[-a, a]$ **,** which represents a shrinking interval since a is being reduced linearly. The Shrinking Encircling mechanism can then be achieved using equation 3.3.6:

$$\vec{X}(t+1) = \overrightarrow{X^*} - \vec{A} \cdot \vec{D} \qquad 3.3.6$$

Where $\overrightarrow{X^*}$ is the prey position which denotes the best solution found so far.

- *Spiral updating Position mechanism:*

In this mechanism, each individual updates its position following a spiral-shaped path around the prey position $\overrightarrow{\boldsymbol{X}^*}$ according to the following equation:

$$\vec{X}(t+1) = \overrightarrow{D'} \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X^*} \qquad 3.3.7$$

Where D' is the distance between the whale and the prey location:

$$\overrightarrow{D'} = \left|\overrightarrow{X^*} - \vec{X}(t)\right| \qquad 3.3.8$$

b is a constant for defining the shape of the logarithmic spiral, and $\boldsymbol{l}$ is a random number in the interval$[-1,1]$.
The bubble net hunting phase is then accomplished by choosing one of the two modes stated above, depending on a probability of selection (50%) as follows:

$$\vec{X}(t+1) = \begin{cases} \overrightarrow{X^*} - \vec{A} \cdot \vec{D} & if \quad p < 0.5 \\ \overrightarrow{D'} \cdot e^{bl} \cdot \cos(2\pi l) + \overrightarrow{X^*} & if \quad p \geq 0.5 \end{cases} \qquad 3.3.9$$

Where p is a random number in the range $[0,1]$.This behavior is depicted in figure 30.b.

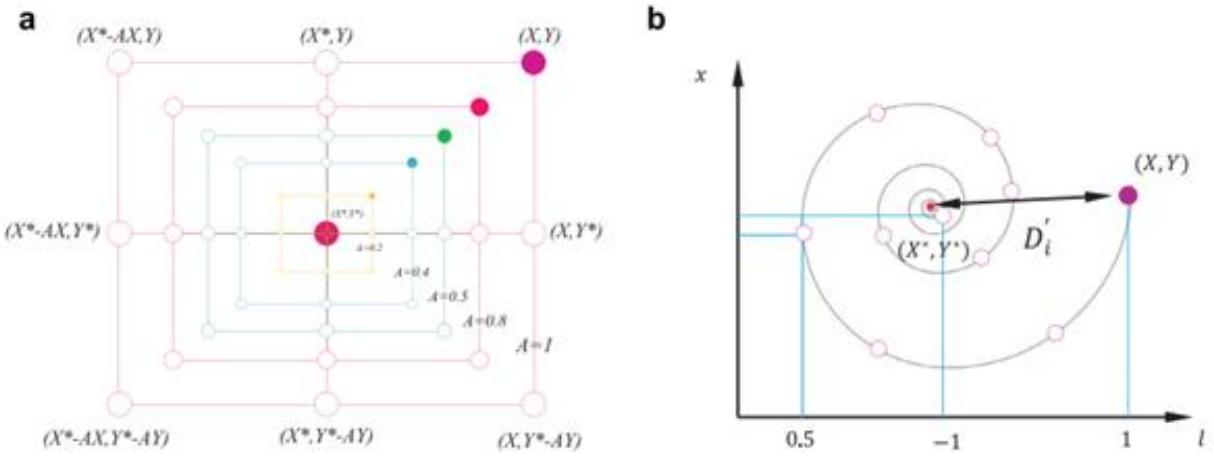

Figure 30:Bubble-net search mechanism implemented in WOA (X is the best solution obtained so far): (a) shrinking encircling mechanism and (b) spiral updating position [20].*

C- Encircling the prey:

In this stage, the search agents will update their positions towards the prey location $\overrightarrow{\boldsymbol{X}^*}$. This phase is performed when $|\vec{A}| < 1$, and it is modeled using the following equation:

$$\overrightarrow{X(t+1)} = \overrightarrow{X^*} - \vec{A} \cdot \vec{D} \qquad 3.3.10$$

It is worth noting that in this mode of operation which can be interpreted as the exploitation phase, is different from the exploration phase in that the whales' positions are updated in reference to the optimum solution rather than a randomly chosen whale position.

The process steps of the WOA are presented in the following pseudo-code :

Initialize the whales population $X_i(i = 1,2, \ldots \ldots, n)$
Calculate the fitness of each search agent
$X^* = the\ best\ search\ agent$

While ($t < Maximum\ number\ of\ iterations\ T$)
For each search agent
Update $a\ , A, C, I\ and\ p$
if1 (P < 0.5)
if2 ($|A| < 1$)
Update the position of the current search agent by Eq. (3.3.6)
else if2 ($|A| \geq 1$)
Select a random search agent (X_{rand})
Update the position of the current search agent by Eq. (3.3.1)
end if2
else if1 (p ≥ 0.5)
update the position of the current search agent by the Eq. (3.3.7)
end if1
end for
check if any search agent goes beyond the search space and amend it
calculate the fitness of each search agent
update X^* if there is a better solution
t= t+1
end while
return X^*

3.3.3 WOA based MPPT:

The WOA starts with the initialization of a vector containing a set of *N* duty cycle solutions. Each individual power is evaluated and the prey position which represents the best solution is obtained.

The algorithm then in each iteration, selects one of the two modes of equation 3.3.9 based on the value of the randomly generated number. If the *Spiral Updating Position mechanism* is chosen, then the duty cycles are updated by either using a randomly selected search agent from the population (equation 3.3.1) or using the prey position $dbest$ (equation 3.3.6) based on the value of A. If in the other hand the *shrinking Encircling mechanism* is selected, then equation 3.3.7 will be employed. When the maximum number of iterations is achieved, the optimum solution $dbest$ is sent to the power converter.

The algorithm has to be reinitialized whenever a change in weather conditions is detected, according to the following inequality constraint:

$$\frac{\left|P_{pv-new} - P_{pv-previous}\right|}{P_{pv-previous}} \geq C \qquad 3.3.11$$

As we have mentioned before, the original paper suggests that the value of a is reduced linearly within the range[2,0], moreover the exploration and exploitation phases are selected according to the value of A (Exploration takes place when $|\vec{A}|> 1$ whereas the exploitation is performed when $|\vec{A}|< 1$). However, we have found that this parameter setting is not very suitable in MPPT applications, since it often drives the updated duty cycles to be in the vicinity of the search space bounds (d_{min} and d_{max}). In this work, it is suggested to decrease a over the interval[1,0]. This modification will solve this issue. The critical value that will allow making a transition from Exploration to Exploitation will then selected to be: $|\vec{A}|$ =0.5. The flowchart illustrating the implementation of WOA in maximum power harvesting is shown in figure 31.

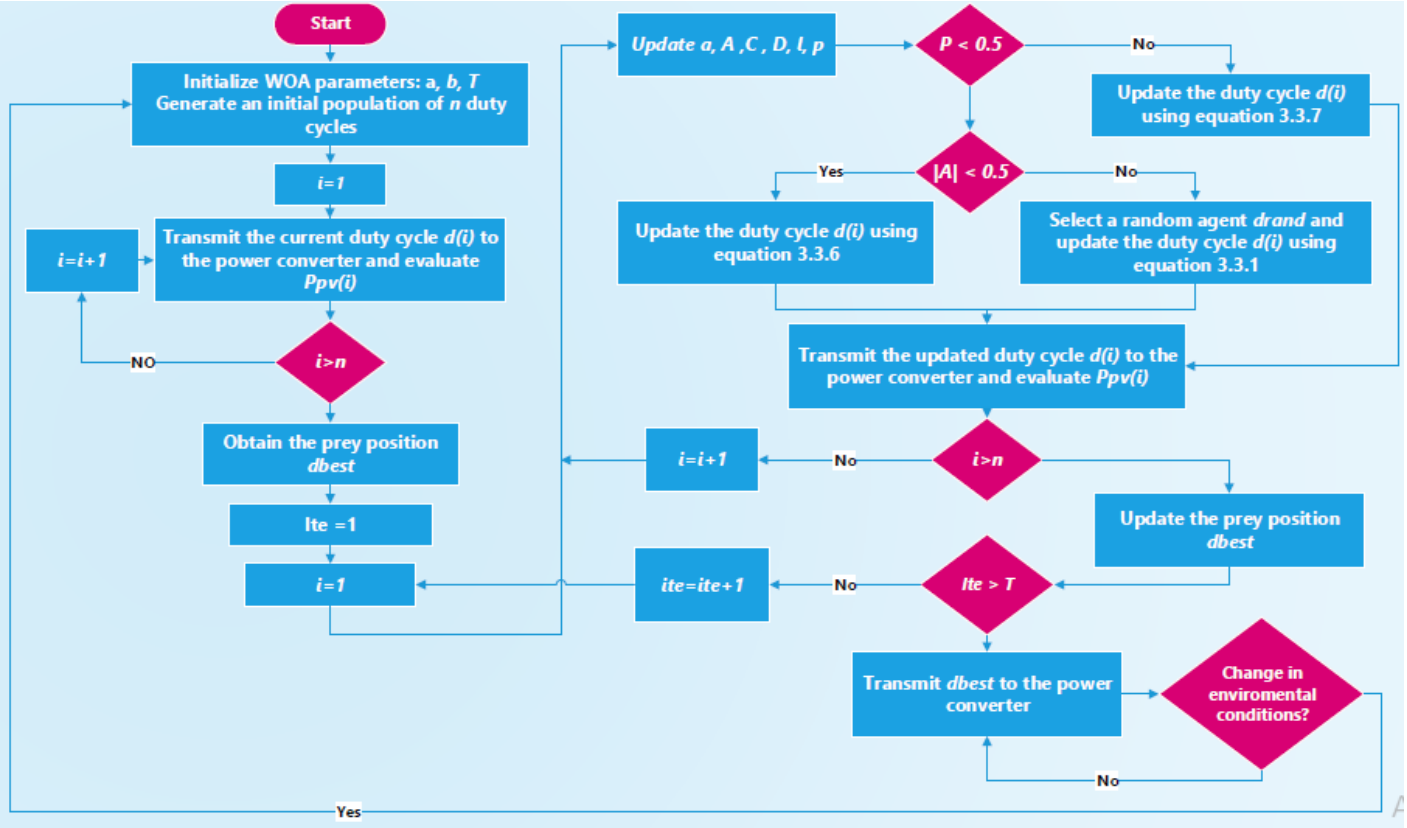

Figure 31:WOA based MPPT.

3.4 Grey Wolf Optimization Algorithm:

3.4.1 Inspiration:

Grey Wolf Optimization (GWO) is a swarm intelligence technique inspired from the hunting behavior and the leadership hierarchy of grey wolves in nature [22]**.** In GWO, pack's members are divided into four groups based on the type of the wolf's role that help in advancing the hunting process: the alphas (α), the betas(β), the deltas(δ), and the omegas(ω) as shown in Fig. 32. The

alphas are the most powerful and the dominant agents in leading the group, the betas represent the secondary wolves that assist the alphas in making decisions whereas the Omegas have the lowest rank. If a wolf is neither an alpha nor a beta, or an omega, it is considered as a delta (sub-ordinate). Another behavior of grey wolfs is their appealing group hunting strategy, which consists of a set of phases:

- Tracking, chasing, and approaching the prey.
- Pursuing, encircling, and harassing the prey until it stops moving
- Attacking the prey.

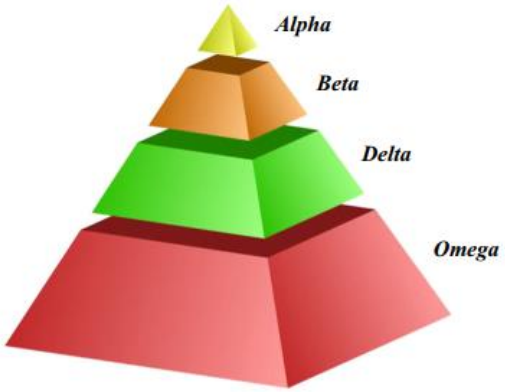

Figure 32:Social Hierarchy of Grey Wolves.

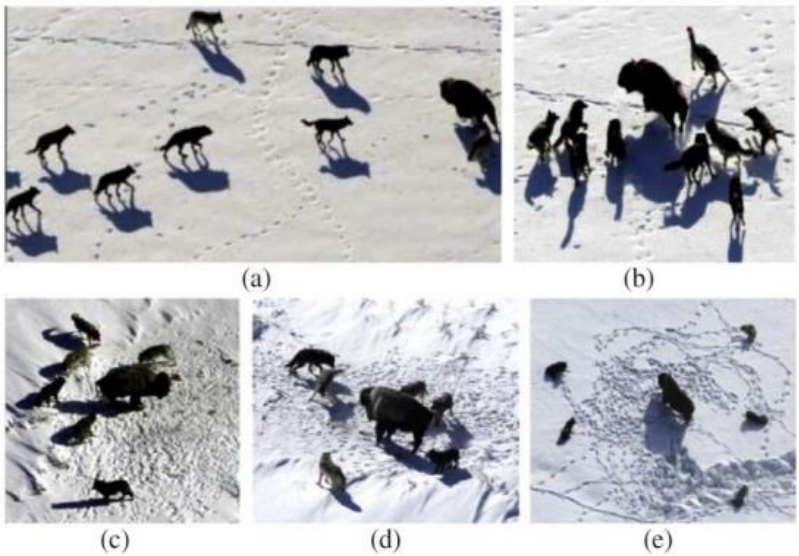

Figure 33:Hunting behavior of grey wolves: (a-c) chasing, approaching and tracking prey; (d) encircling; (e) stationary situation and attack.

3.4.2 Mathematical Modeling:

A - Social hierarchy:

In order to mathematically model the social hierarchy of grey wolves, the three best-so-far solutions are considered to be the alpha (α), beta(β), the delta(δ) agents, where alpha represents the fittest solution, beta(β) and delta(δ) are the second and third fittest agents respectively.

B - Encircling the Prey:

In order to model the encircling strategy, the following equations are used:

$$\overrightarrow{D} = \left|\overrightarrow{C}.\overrightarrow{X}_p(t) - \overrightarrow{X}(t)\right| \quad 3.4.1$$

$$\overrightarrow{X}(t+1) = \overrightarrow{X}_P(t) - \overrightarrow{A}.\overrightarrow{D} \quad 3.4.2$$

where t indicates the current iteration, $\overrightarrow{A}$ and $\overrightarrow{C}$ are coefficient vectors, $\overrightarrow{X}_P$ is the position vector of the prey, and $\overrightarrow{X}$ indicates the individual position vector .

The vectors $\overrightarrow{A}$ and $\overrightarrow{C}$. are calculated as follows:

$$\overrightarrow{A} = 2\vec{a}.\overrightarrow{r_1} - \vec{a} \quad 3.4.3$$

$$\overrightarrow{C} = 2.\overrightarrow{r_2} \quad 3.4.4$$

In Equations 3.4.3 and 3.4.4, $\overrightarrow{r_1}$and $\overrightarrow{r_2}$are random vectors in the interval [0,1]. Whereas $\vec{a}$ is a linearly decreased parameter from 2 to 0.

C - Hunting:

During the hunting process, the GWO assumes that the alpha, beta and delta agents have better knowledge about the promising regions of the prey locations. The search process is then guided by these agents using the following equations:

$$\overrightarrow{D_\alpha} = \left|\overrightarrow{C_1}.\overrightarrow{X_\alpha} - \overrightarrow{X}\right| \quad 3.4.5$$

$$\overrightarrow{D_\beta} = \left|\overrightarrow{C_2}.\overrightarrow{X_\beta} - \overrightarrow{X}\right| \quad 3.4.6$$

$$\overrightarrow{D_\delta} = \left|\overrightarrow{C_3}.\overrightarrow{X_\delta} - \overrightarrow{X}\right| \quad 3.4.7$$

$$\overrightarrow{X_1} = \overrightarrow{X_\alpha} - \overrightarrow{A_1}.(\overrightarrow{D_\alpha}) \quad 3.4.8$$

$$\overrightarrow{X_2} = \overrightarrow{X_\beta} - \overrightarrow{A_2}.(\overrightarrow{D_\beta}) \quad 3.4.9$$

$$\overrightarrow{X_3} = \overrightarrow{X_\delta} - \overrightarrow{A_3}.(\overrightarrow{D_\delta}) \quad 3.4.10$$

$$\overrightarrow{X}(t+1) = \frac{(\overrightarrow{X_1} + \overrightarrow{X_2} + \overrightarrow{X_3})}{3} \qquad 3.4.11$$

Where $\overrightarrow{X_\alpha}$, $\overrightarrow{X_\beta}$, $\overrightarrow{X_\delta}$ indicates the position vector of the alpha, beta and delta agents respectively,

Figure 34 demonstrates how a search agent updates its position according to alpha, beta, and delta in a 2D search space. It can be observed that the final position would be in a random place within a circle which is defined by$\overrightarrow{X_\alpha}$, $\overrightarrow{X_\beta}$, $\overrightarrow{X_\delta}$ in the search space. In other words, alpha, beta, and delta estimate the position of the prey, and other wolves updates their positions randomly around the prey [22].

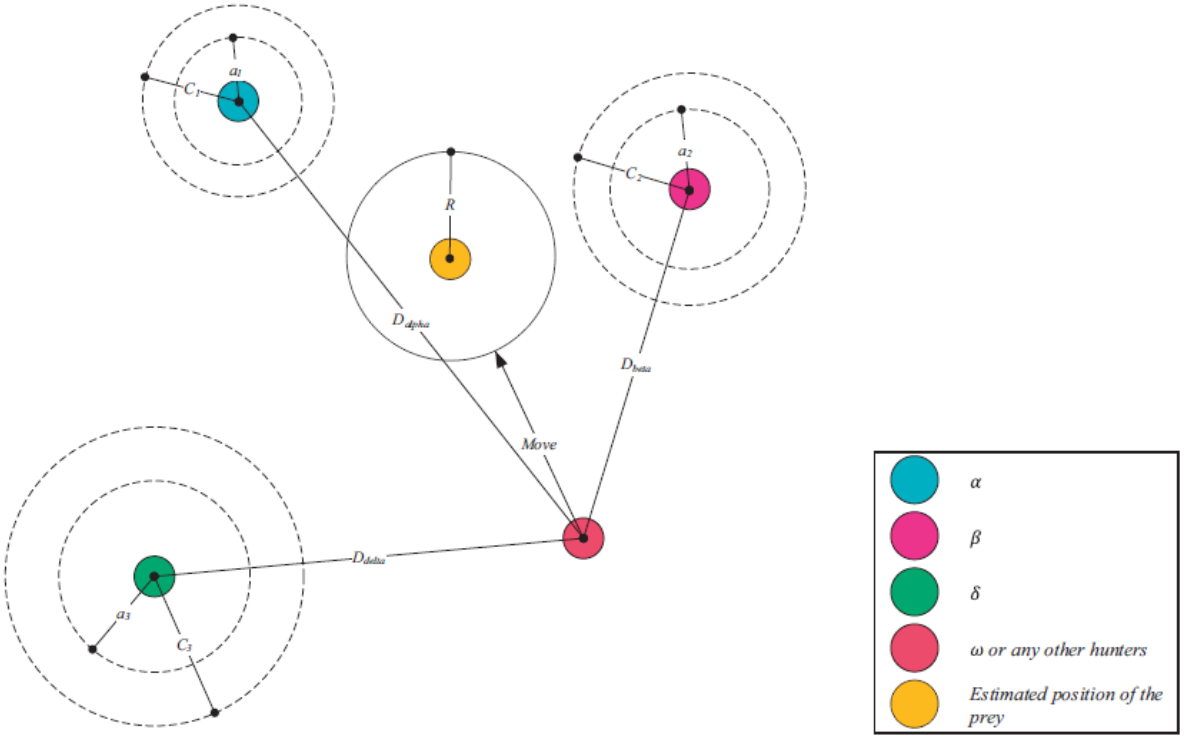

Figure 34:Position Updating in GWO.

D - Exploration and Exploitation:

The two main parameters that control the balance between exploration and exploitation are the parameters C and A.

With A being decreased throughout the optimization process (since a is linearly reduced over [0,2]), half of the iterations are devoted to exploration ($|A| \geq 1$) whereas the others are dedicated to exploitation ($|A| < 1$). The C vector contains random values in [0, 2]. This component provides random weights for the prey in order to stochastically emphasize C >1 or deemphasize C <1its

effect in defining the distance in Eq. 3.4.1. This assists GWO to show a more random behavior throughout optimization, favoring exploration and local optima avoidance [23]. The pseudo code of the GWO algorithm is given below:

Initialize the grey wolf population $X_i(i = 1,2, \ldots\ldots, n)$
Initialize A, a and C
Calculate the fitness of each search agent
X_α= the best search engine
X_β= the second best search engine
X_δ= the third best search engine
While (t< *Max number of iterations T*)
    For each search agent
        Update the position of the current search agent by equation (3.4.11)
    End for
Update a, A and C
    Calculate the fitness of all search agent
    Update X_α, X_β and X_δ
    t = t+1
end while
return X_α

3.4.3 GWO based MPPT:

The flowchart illustrating the Grey Wolf Optimization based MPP tracker is shown in figure 35:

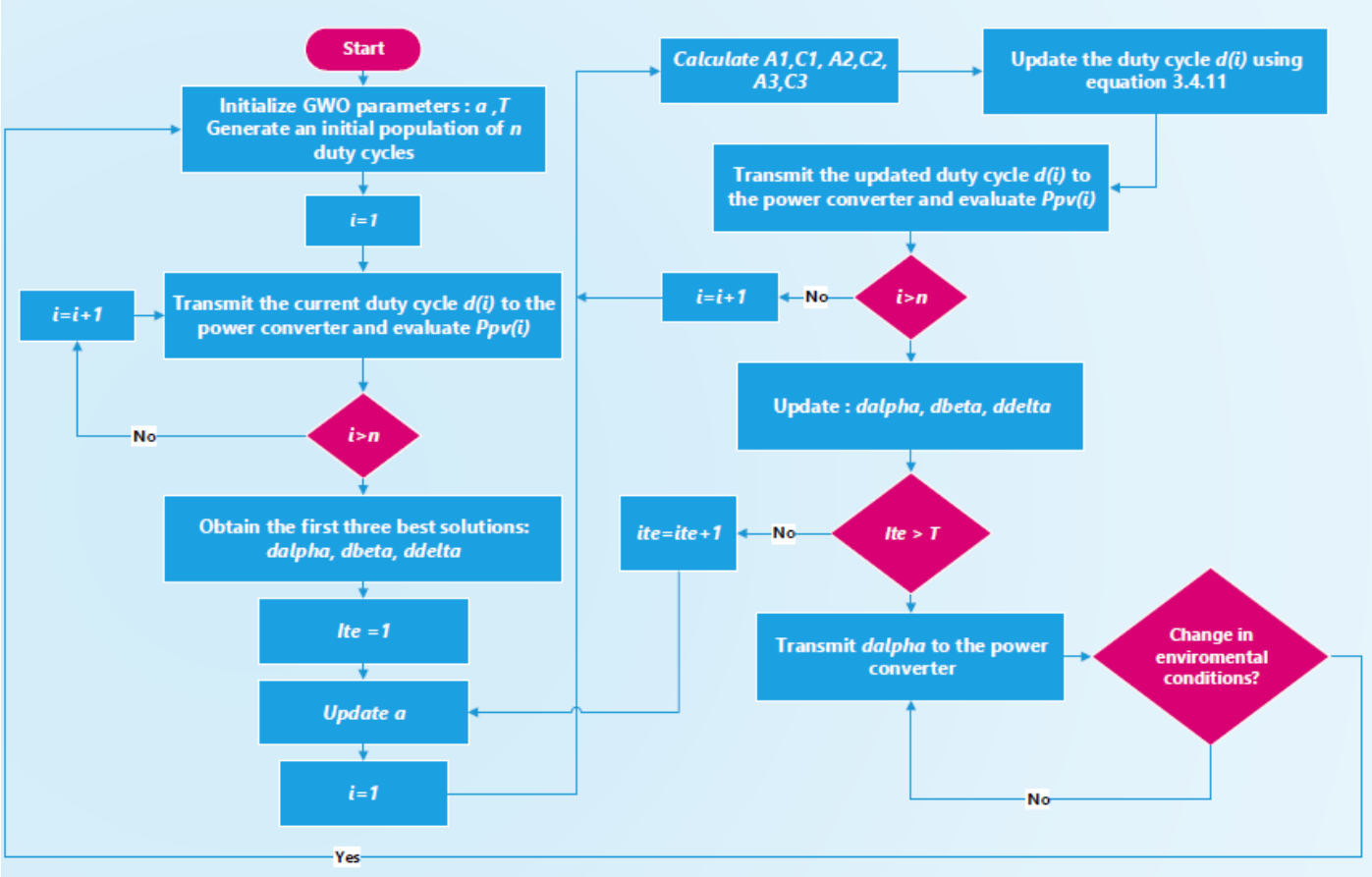

Figure 35:GWO based MPPT

3.5 Wind Driven Optimization (WDO):

3.5.1 Inspiration and Modeling:

The inspiration for Wind Driven Optimization comes from the earth's atmosphere, where horizontal differences in atmospheric pressure cause air to move and hence wind to blow [24]. Since temperature differences lead to variations in air density and air pressure at different locations, horizontal unbalances in air pressure will drive the air to move from high pressure regions to low pressure regions at a certain velocity until air pressure is balanced [25]. The derivation of the WDO technique starts with Newton's second law of motion applied to the movement of air parcels:

$$\rho\vec{\alpha} = \sum \vec{F}_i \qquad 3.5.1$$

where $\vec{\alpha}$ is the acceleration, ρ is the air density for an infinitesimal air parcel, and $\vec{F}_i$ are the forces acting on the air parcel.

The main forces acting on air parcels producing wind are essentially:

a- Pressure gradient force $\vec{F}_{PG}$:

The pressure gradient force is the central force that causes wind to blow and it is the direct result of variations in horizontal air pressure. The pressure gradient force is directed from high pressure zones to low pressure zones and it expressed as:

$$\vec{F}_{PG} = -\nabla\vec{P}\delta \qquad 3.5.2$$

Where $\nabla\vec{P}$is the pressure gradient and it is defined as the difference in pressure between two places over the distance between them.

b- ***The gravitational force $\vec{F}_G$***:

The gravitational force can be treated as a vertical force directed towards the earth surface, or simply the origin of the rectangular coordinate system and it is given by:

$$\vec{F}_G = m\vec{g} \qquad 3.5.3$$

Considering the air has finite mass and finite volume $\boldsymbol{\delta V}$, equation 3.5.3 can be rewritten as

$$\vec{F}_G = \rho\delta V\vec{g} \qquad 3.5.4$$

c- The friction force $\vec{F}_F$

The friction force acts to oppose the motion started by the pressure gradient force [26] and it is expressed as:

$$\vec{F}_F = -\rho\alpha\vec{u} \qquad 3.5.5$$

Where $\vec{u}$ denotes the air parcel velocity vector.

d- The Coriolis force $\vec{F}_C$:

The *Coriolis* force causes the deflection of the wind from its existing path in the atmosphere [26] where the amount of deflection Ω is due the earth's rotation. This force is given in equation 3.5.6:

$$\vec{F}_C = -2\Omega\vec{u} \qquad 3.5.6$$

The four forces are summed up accordingly with equation 3.5.1:

$$\rho\vec{\alpha} = \rho\delta V\vec{g} + \left(-\nabla\vec{P}\delta V\right) + (-\rho\alpha\vec{u}) + (-2\Omega\overrightarrow{u}) \quad 3.5.7$$

Since $\overrightarrow{\alpha}$ can be written as $\frac{\Delta\vec{u}}{\Delta t}$, and setting $\Delta t = 1$ for the sake of simplicity and letting $\delta V = 1$ for an infinitesimal air parcel, equation 3.5.7 reduces to:

$$\rho\Delta\vec{u} = \rho\vec{g} + \left(-\nabla\vec{P}\right) + (-\rho\alpha\vec{u}) + (-2\Omega\overrightarrow{u}) \quad 3.5.8$$

The ideal gas law relates air pressure with the air density and temperature by:

$$P = \rho RT \quad 3.5.9$$

Where P is the air parcel pressure, R is the universal gas constant and T is the temperature. Equation 4.5.8 can then be rewritten as:

$$\Delta\vec{u} = \vec{g} + \left(-\nabla\vec{P}\frac{RT}{P_{cur}}\right) + (-\alpha\vec{u}) + (\frac{-2\Omega\mathrm{RT}\vec{u}}{P_{cur}}) \quad 3.5.10$$

Where P_{cur} is the current location pressure.

The WDO technique updates the air parcels velocity and position at each iteration during the whole optimization process. The change in velocity can be expressed as:

$$\Delta\vec{u} = \vec{u}(t+1) - \vec{u}(t) \quad 3.5.11$$

Since the gravitational force acts to pull air parcels from their current position into the center of the coordinate system, the acceleration vector $\vec{g}$is decomposed into:

$$\vec{g} = |g|(o - x_{cur}) \quad 3.5.12$$

The pressure gradient vector $\nabla\vec{P}$ represents the force that drives air parcels, caused by variations of pressure in two different locations. This vector can be broken down into:

$$\nabla\vec{P} = \left|P_{opt} - P_{cur}\right|\left(x_{opt} - x_{cur}\right) \quad 3.5.13$$

Where P_{opt} is the optimum pressure that corresponds to location x_{opt} found so far by the population.

The *Coriolis* force is replaced by the velocity influence of another randomly chosen dimension of the same parcel, and all other coefficients are combined into a single term:

$$c = -2|\Omega|\mathrm{RT} \quad 3.5.14$$

Updating equation 3.5.8 using equations 3.5.9, 3.5.10, 3.5.11, 3.5.12, 3.5.13:

$$\vec{u}(t+1) = (1-\alpha)\vec{u}(t) - g\vec{x}(t) + \frac{RT}{P_{cur}}\left|P_{opt} - P_{cur}\right|\left(x_{opt} - \vec{x}(t)\right) + \left(\frac{\overrightarrow{cu_{dim}^{other}}}{P_{cur}}\right) \quad 3.5.15$$

Sometimes where the pressure is very high, the updated velocities will have large values and the algorithm efficiency will be affected negatively [30]. To avoid such situations, the actual pressure values are replaced by a ranking integer i, such that the best solution has a rank of $i = 1$. Equation 3.5.15 will then be reformed as:

$$\vec{u}(t+1) = (1-\alpha)\vec{u}(t) - g\vec{x}(t) + RT\left|1 - \frac{1}{i}\right|\left(x_{opt} - \vec{x}(t)\right) + \left(\frac{\overrightarrow{cu_{dim}^{other}}}{i}\right) \quad 3.5.16$$

Where $\vec{u}(t)$is limited to a specified range $[-V_{max}, V_{max}]$ such that :

$$u(t) = \begin{cases} V_{max}, & u(t) > V_{max} \\ -V_{max}, & u(t) < -V_{max} \end{cases} \quad 3.5.17$$

Air parcels positions are updated using:

$$\vec{x}(t+1) = \vec{x}(t) + \vec{u}(t+1) \quad 3.5.18$$

It should be pointed out that the WDO technique allows the air parcels movement to be within the range [-1, 1], such that:

$$\vec{x}(t) = \begin{cases} 1, & x(t) > 1 \\ -1, & x(t) < -1 \end{cases} \quad 3.5.19$$

The resulting particles positions are then mapped back into the search space of the given optimization problem.

Equations 3.5.16 and 3.5.18 represent the general framework of the Wind Driven Optimization algorithm.

3.5.2 WDO based MPPT:

Similar to the other metaheuristic algorithms, in the WDO, proper tuning of the control parameters is necessary for effective performance, this include:

- Population size : N
- RT coefficient
- Friction coefficient α
- Gravitational constant: g

- The c constant
- Velocity limits V_{max} and V_{min}

In order to guarantee convergence of the algorithm, we introduced some modifications to the original version of the WDO such that:

- The interval of allowable velocities $[-V_{max}\ ,\ V_{max}]$ is as follows:

$$V_{max} = V_{max0}\ (1 - \frac{t}{T})$$

- a and c are also updated within a specified range along the whole optimization process.

$$a = a_0(1 - \frac{t}{T})$$

$$c = c_0(1 - \frac{t}{T})$$

The chosen settings (V_{max0} , $a_0\ and c_0$) are provided in chapter 4.

Since our application is of one dimension, the velocity term u_{dim}^{other} in equation 3.5.16 will be simply $u(t)$. The WDO process steps in tracking the MPP are summarized below and illustrated in the flowchart of figure 36.

Step1: Parameters Initialization: N, RT, a_0, g, c_0, V_{max0} and Maximum number of iterations T.

Step 2: A solution vector of N air parcels representing a set of duty cycles and a velocity vector of the same length are initialized.

Step 3: The generated duty cycles are transmitted to the boost converter and the corresponding power is evaluated.

Step 4: Obtain the global best duty cycle *Gbest*.

Step 5: Set t=1.

Step 6: Update the population using equations 3.5.16 and 3.5.18 along with equations 3.5.17 and 3.5.19.

Step 7: The updated positions are transformed back to the selected search space [dmin , dmax] andtransmitted to the power converter for power evaluation.

Step 8: Duty cycles and velocities are sorted in descending order according to the obtained power values (The duty cycle with the highest obtained power will be at rank 1)

Step 9: Update *Gbest* if a better solution is obtained and increment t.

Step 10: If $t < T$ or the convergence criterion is not fulfilled, update the control parameters (α , c, V_{max}), and repeat steps 5 through 8. Otherwise go to step 10.

Step 11: Transmit *Gbest* to the power converter.

Step 12: Irradiation change detection: if $\frac{|P_{old}-P_{new}|}{P_{old}} \geq C$ the algorithm is reinitialized, otherwise return to step 10.

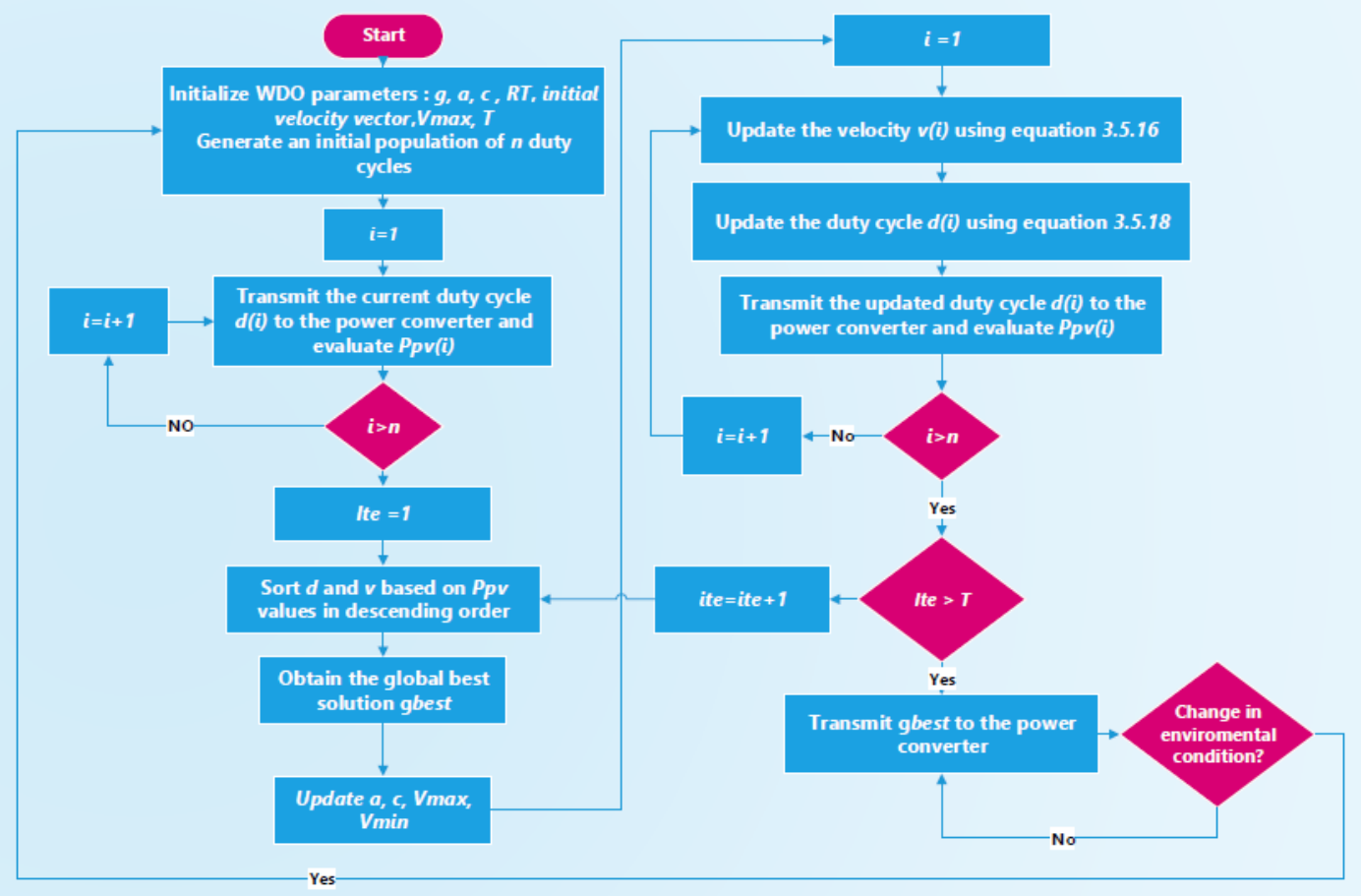

Figure 36:WDO based MPPT.

3.6 Grasshopper Optimization Algorithm:

3.6.1 Inspiration:

Grasshopper optimization algorithm (GOA) is a recent meta-heuristic population-based optimization technique developed in 2017 [28], and it mimics the swarming behaviors of grasshopper insects in nature and their social interaction**.** These insects have two distinct stages in their lifecycle, namely nymph and adulthood. During the first phase, grasshoppers have no wings, which makes them move in slow and small steps. In the other hand, their adulthood stage is characterized by long-range, fast and abrupt movements.

The nymph grasshopper moves like rolling cylinders in millions of numbers. They almost eat all the vegetation which comes in their path during their movement. When they become adult from nymph, they form a swarm in air and then they migrate over very large distances.

The social interaction between grasshoppers is categorized into three zones, repulsion zone, comfort zone, and attraction zone. Figure 37 shows the life cycle and the social interactions between grasshoppers.

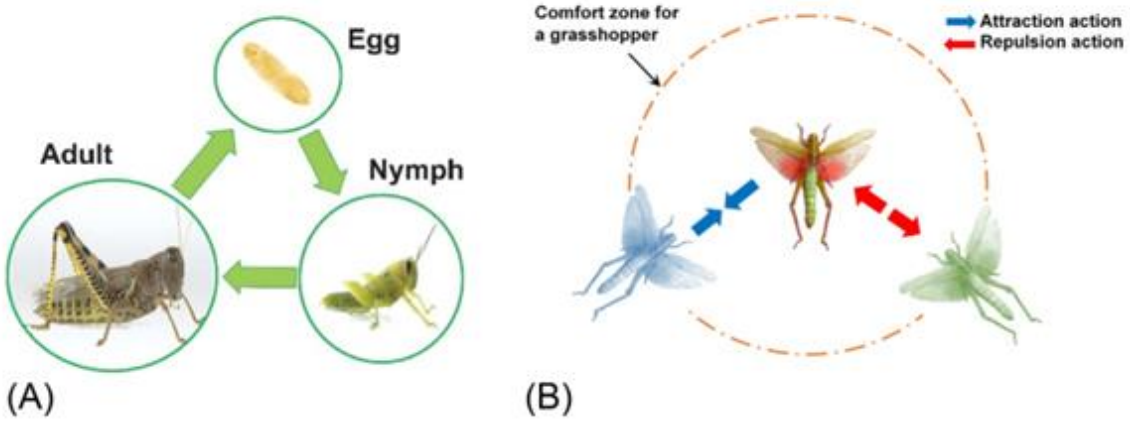

Figure 37:(A) The life cycle of a grasshopper; (B) Social interaction between various grasshoppers [27].

3.6.2 Mathematical Modeling:

The movement of grasshoppers can be simulated using the following equation:

$$X_i = S_i + G_i + W_i \tag{3.6.1}$$

where X_i is the ith grasshopper position, S_i denotes the social factor, G_i denotes the impact of gravity, and W_i denotes the wind's horizontal impact.
A randomized version of Eq. 3.6.1, is given below, where, r1, r2, and r3 are random factors within the range of [0, 1].

$$X_i = r_1 S_i + r_2 G_i + r_3 W_i \tag{3.6.2}$$

The social interaction plays a key role in the movement of grasshoppers, and it is calculated using Eq. 3.6.3, where the *s* function measures the intensity of social interaction:

$$S_i = \sum_{\substack{j=1 \\ j \neq i}}^{N} s(d_{ij}) \times \widehat{d_{ij}} \tag{3.6.3}$$

d_{ij} is the distance between two distinct grasshoppers, while $\widehat{d_{ij}}$ is a unit vector between the *ith* search agent to the *jth* search agent, given as follows:

$$d_{ij} = |X_j - X_i| \quad 3.6.4$$

$$\widehat{d_{ij}} = \frac{X_j - X_i}{d_{ij}} \quad 3.6.5$$

The $s(d_{ij})$function defines the social forces and it is calculated as follows:

$$s(r) = f \times e^{\frac{-r}{l}} - e^{-r} \quad 3.6.6$$

Where f indicates the intensity of attraction, while l denotesthe attractive length scale.
It should be underlined that the function *s* returns values close to zero with distances greater than 10, so that strong force among the grasshoppers with large distances between them cannot be applied. To resolve this problem, distances must be mapped into the interval [1,4].

The G and W components in Eq. 3.6.2 are calculated as follows:

$$G_i = -g\widehat{e_g} \quad 3.6.7$$

$$W_i = k\widehat{e_w} \quad 3.6.8$$

Where g denotes the gravitational constant, $\widehat{e_g}$ represents a unity vector towards the center of the earth, k represents a constant drift, and $\widehat{e_w}$is a unity vector towards the wind direction.
Substituting *S, G*, and *W* in equation.3.6.2, results in:

$$X_i = \sum_{\substack{j=1 \\ j\neq i}}^{N} s(|X_j - X_i|) \times \frac{X_j - X_i}{d_{ij}} - g\widehat{e_g} + k\widehat{e_w} \quad 3.6.9$$

Since nymph grasshoppers land on the ground, their position should not go below a threshold. However, we will not utilize this equation in the swarm simulation and optimization algorithm because it prevents the algorithm from exploring and exploiting the search space around a solution [28]**.** Equation 4.6.9 is then reformed as follows:

$$X_i^m = c\left(\sum_{\substack{j=1 \\ j\neq i}}^{N} c\frac{ub_m - lb_m}{2}(|X_j^m - X_i^m|) \times \frac{X_j - X_i}{d_{ij}}\right) + \widehat{T_m} \quad 3.6.10$$

Where ub_m and lb_m represent the upper and lower bounds in the mth dimension respectively, T_m is the m-dimensional position of the fittest solution, and C is given by Eq. 3.6.11 and it acts as a

decreasing factor within the range $[C_{min}, C_{max}]$ to shrink the comfort area, repulsion area, and attraction area:

$$C = C_{max} - t\frac{C_{max} - C_{min}}{T} \quad 3.6.11$$

The pseudo code of the GOA algorithm is given below:

Initialize the swarm $X_i (i = 1,2, \ldots \ldots, n)$

Initialize C_{min}, C_{max} , and *Maximum number of iterations T*

Calculate the fitness of each search agent

T_m = the best search agent

While ($t < Maximum\ number\ of\ iterations\ T$)

Update C using equation (3.6.11)

for each search agent

Normalize the distances between grasshoppers in $[1,4]$.

Update the position of the current search agent by the equation (3.6.10)

Bring the current search agent back if it goes outside the boundaries

end for

Update T_m if there is a better solution

$t = t + 1$

End while

3.6.3 GOA based MPPT:

The operating flowchart of the GOA based MPP tracker is shown in figure 38:

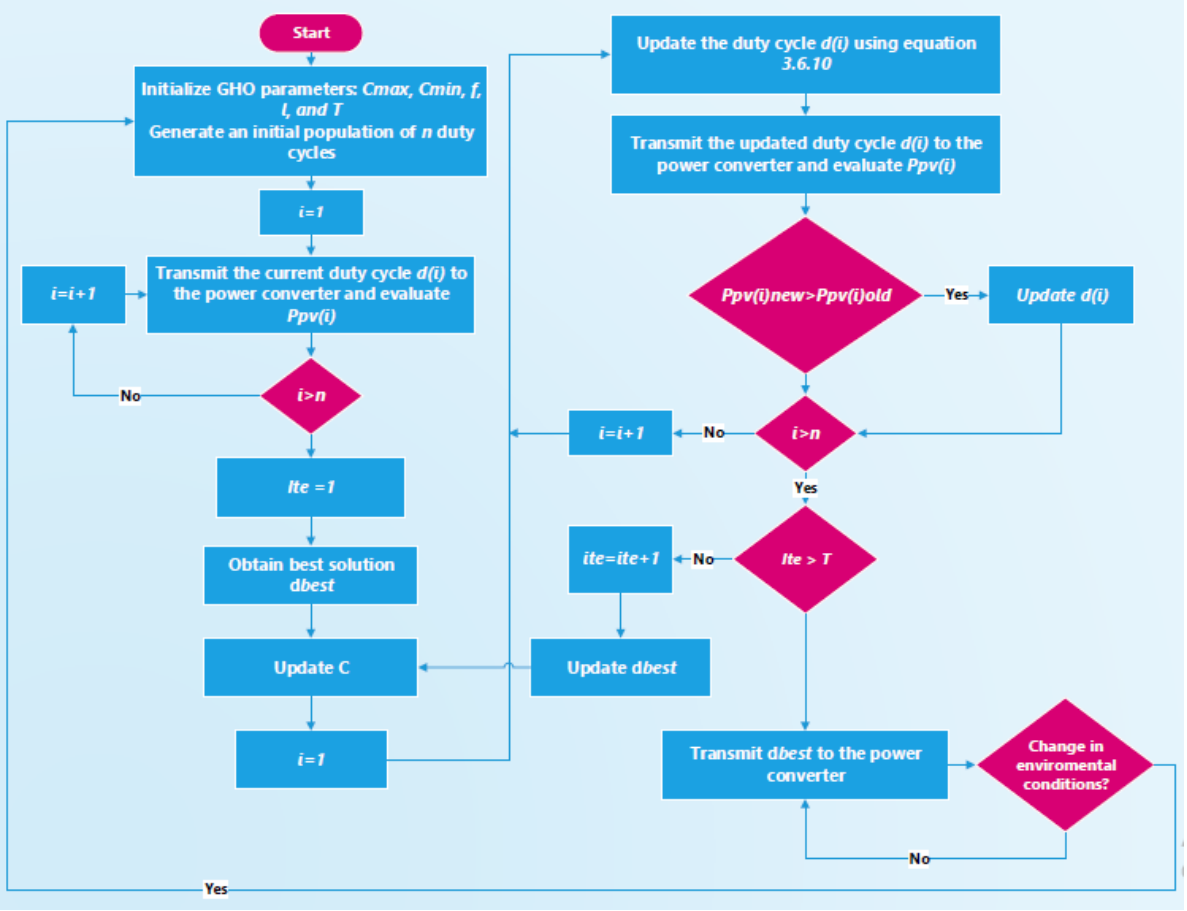

Figure 38: GOA based MPPT

3.7 Equilibrium Optimizer:

3.7.1 Inspiration:

The Equilibrium Optimizer is a novel algorithm proposed by Afshin Faramarzi, Mohammad Heidarinejad, Brent Stephens and Seyedali Mirjaliliin 2019 [30] The inspiration for the EO technique is a simple well-mixed dynamic mass balance on a control volume, in which a mass balance equation is used to describe the concentration of a nonreactive constituent in a control volume, as a function of its various source and sink mechanisms. The mass balance equation provides the underlying physics for the conservation of mass entering, leaving, and generated in a

control volume. A first-order ordinary differential equation expressing the generic mass-balance equation [29], in which the change in mass in time is equal to the amount of mass that enters the system plus the amount being generated inside minus the amount that leaves the system, as described in [30].

$$V\frac{dC}{dt} = QC_{eq} - QC + G \qquad 3.7.1$$

Where C denotes the concentration inside the control volume, $\frac{dC}{dt}$ is the rate of change of mass in the control volume, Q is the volumetric flow rate into and out of the control volume, C_{eq}represents the concentration at an equilibrium state in which there is no generation inside the control volume, and G is the mass generation rate inside the control volume.

A steady equilibrium state is reached when $\frac{dC}{dt}$ becomes zero. Rearranging equation 3.8.1, the concentration in the control volume (**C**) as a function of time (**t**) is obtained:

$$C = C_{eq} + (C_o - C_{eq})F + \frac{G}{\lambda V}(1 - F) \qquad 3.7.2$$

Where $\lambda = \frac{Q}{V}$ is the turnover rate.

$t_0\ and\ C_o$and are the initial start time and initial concentration respectively, and F will be discussed later.

3.7.2 EO Modeling and Process steps:

A- Initialization:

Similar to the other evolutionary algorithms, in the EO technique a population of *n* individuals is randomly generated where each individual (solution) with its concentration (position) acts as a search agent.

B- Equilibrium Pool and Candidates:

The equilibrium state is the final convergence state of the algorithm, which is desired to be theglobal optimum solution. The algorithm uses a set of the best solutions found so far during the optimization process to update the search agent's concentration, these best-so-far solutions are named *equilibrium candidates*. The choice of the set size is arbitrary and depends on the type of the optimization problem, however, the developers of this algorithm pointed out that using less than four candidates degrades the performance of the method in multimodal and composition functions but will improve the results in unimodal functions. More than four candidates will have the opposite effect. These four candidates help EO to have a better exploration capability, while the exploitation is enhanced by adding another particle whose concentration is the arithmetic mean to the set [30].

The whole set embraces five individuals, and it is used to construct a vector nominated as the *Equilibrium pool*:

$$\overrightarrow{C_{eq,Pool}} = \{\overrightarrow{C_{eq(1)}}, \overrightarrow{C_{eq(2)}}, \overrightarrow{C_{eq(3)}}, \overrightarrow{C_{eq(4)}}, \overrightarrow{C_{eq(avg)}}\} \quad 3.7.3$$

The equilibrium pool is used to update the population by randomly selecting one of the equilibrium candidates in each iteration for each individual.

C- Exponential Term F:

The exponential term given in equation 3.7.4 will assist EO in having a reasonable balance between diversification and intensification. Since the turnover rate $\boldsymbol{\lambda}$ can vary with time in a real control volume, it is assumed to be a random vector in the interval of [0,1]

$$F = e^{-\lambda(t-t_0)} \quad 3.7.4$$

Where t is reduced following equation 3.7.5:

$$t = (1 - \frac{ite}{T})^{(a_2 ite/T)} \quad 3.7.5$$

T is the maximum number of iterations and a_2 is a constant value used to manage the exploitation ability.

In order to guarantee convergence, t_0 is also reduced over the whole optimization process using equation 3.7.6

$$t_0 = \frac{1}{\lambda}\ln\left(-a_1 sign(r-0.5)\left[1-e^{-\lambda t}\right]\right) + t \quad 3.7.6$$

Where r is a random vector in the range [0,1], a_1 is a constant value used to manage exploration ability.

Substituting equation 3.7.6 into equation 3.7.4, a simpler version for F is obtained:

$$F = a_1 sign(r-0.5)[e^{-\lambda t} - 1] \quad 3.7.7$$

D- Generation Rate G:

The generation rate is given in equation 3.7.8 and it is one of the most important terms in the proposed algorithm to provide the exact solution by improving the exploitation phase.

$$\vec{G} = \overrightarrow{G_o}\vec{F} \quad 3.7.8$$

Where $\overrightarrow{G_o}$is computed using:

$$\overrightarrow{G_o} = \overrightarrow{GCP}(\overrightarrow{C_{eq}} - \vec{\lambda}\vec{C}) \quad 3.7.9$$

$\overrightarrow{GCP}$ is the Generation Rate Control Parameter and it is used to decide whether the G factor will contribute to the concentration updating process of a certain particle or not, this mechanism is modeled in equation 3.7.10:

$$\overrightarrow{GCP} = \begin{cases} 0.5r_1 r_2 \geq GP \\ 0 \quad otherswise \end{cases} \quad 3.7.10$$

Where $\boldsymbol{r_1}$and $\boldsymbol{r_2}$are random numbers in [0,1] and GP is recommended to be 0.5 to provide a good balance between exploration and exploitation.

The general framework of the EO is then:

$$\vec{C}_i^{it+1} = \vec{C}_{eq}^{it} + (\vec{C}_i^{it} - \vec{C}_{eq}^{it})\vec{F}_i^{it} + \frac{\vec{G}_i^{it}}{\vec{\lambda}_i^{it} V}(1 - \vec{F}_i^{it}) \quad 3.7.11$$

The detailed pseudo-code of the EO technique is shown be below

Initialize the particle's populations $i = 1, \ldots\ldots, n$
Assign equilibrium candidates' fitness a large number
Assign free parameters a_1 , a_2, *GP* = 0.5
While (i*te < Maximum number of iterations)*
For$i = 1{:}n$
Calculate fitness ofi^{th} particle
If fit ($\overrightarrow{C_\iota}$) < fit($\overrightarrow{C_{eq\,1}}$)
Replace $\overrightarrow{C_{eq\,1}}$ with $\overrightarrow{C_\iota}$ and fit($\overrightarrow{C_{eq\,1}}$) with fit($\overrightarrow{C_\iota}$)
Else if fit ($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,1}}$) & fit($\overrightarrow{C_\iota}$) < fit($\overrightarrow{C_{eq\,2}}$)
Replace $\overrightarrow{C_{eq\,2}}$ with $\overrightarrow{C_\iota}$ and fit($\overrightarrow{C_{eq\,2}}$) with fit($\overrightarrow{C_\iota}$)
Else if fit($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,1}}$) & fit($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,2}}$) & fit($\overrightarrow{C_\iota}$) < fit($\overrightarrow{C_{eq\,3}}$)
Replace $\overrightarrow{C_{eq\,3}}$ with $\overrightarrow{C_\iota}$ and fit($\overrightarrow{C_{eq\,3}}$) with fit($\overrightarrow{C_\iota}$)
Else if fit($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,1}}$) & fit($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,2}}$) & fit($\overrightarrow{C_\iota}$) > fit($\overrightarrow{C_{eq\,3}}$)& fit($\overrightarrow{C_\iota}$) < fit($\overrightarrow{C_{eq\,4}}$)
Replace $\overrightarrow{C_{eq\,4}}$ with $\overrightarrow{C_\iota}$ and fit($\overrightarrow{C_{eq\,4}}$) with fit($\overrightarrow{C_\iota}$)
End **if**
End **For**

$\overrightarrow{C_{ave}} = (\overrightarrow{C_{eq\,1}} + \overrightarrow{C_{eq\,2}} + \overrightarrow{C_{eq\,3}} + \overrightarrow{C_{eq\,4}})/4$;

Construct the equilibrium pool $\overrightarrow{C_{eq\,pool}} = \{\overrightarrow{C_{eq\,1}}, \overrightarrow{C_{eq\,2}}, \overrightarrow{C_{eq\,3}}, \overrightarrow{C_{eq\,4}}, \overrightarrow{C_{ave}}\}$

Accomplish memory saving (if *ite > 1*)

Assign $t = (1 - \frac{ite}{max\_iter})^{(a2\frac{ite}{max\_iter})}$ Eq.(3.7.5)

For $i = 1:n$

Randomly choose one candidate from the equilibrium pool (vector)

Generate random vectors of $\vec{\lambda}, \vec{r}$

Construct $\vec{F} = a_1\, sign(\vec{r} - 0.5)\left[e^{-\vec{\lambda}\, t} - 1\right]$ Eq.(3.7.7)

Construct $\overrightarrow{GCP} = \begin{cases} 0.5r1 & r2 \geq GP \\ 0 & r2 < GP \end{cases}$ Eq.(3.7.10)

Construct $\overrightarrow{G_0} = \overrightarrow{GCP}\,(\overrightarrow{C_{eq}} - \vec{\lambda}\vec{C})$ Eq.(3.7.9)

Construct $\vec{G} = \overrightarrow{G_0}\,.\vec{F}$ Eq.(3.7.8)

Update Concentrations $\vec{C} = \overrightarrow{C_{eq}} + (\vec{C} - \overrightarrow{C_{eq}}).\vec{F} + \frac{\vec{G}}{\vec{\lambda}\, V}(1 - \vec{F})$

End **For**

ite = ite +1

End **while**

3.7.3 EO based MPPT:

The operating flowchart of the EO for Maximum Power Point Tracking is depicted in figure 39.

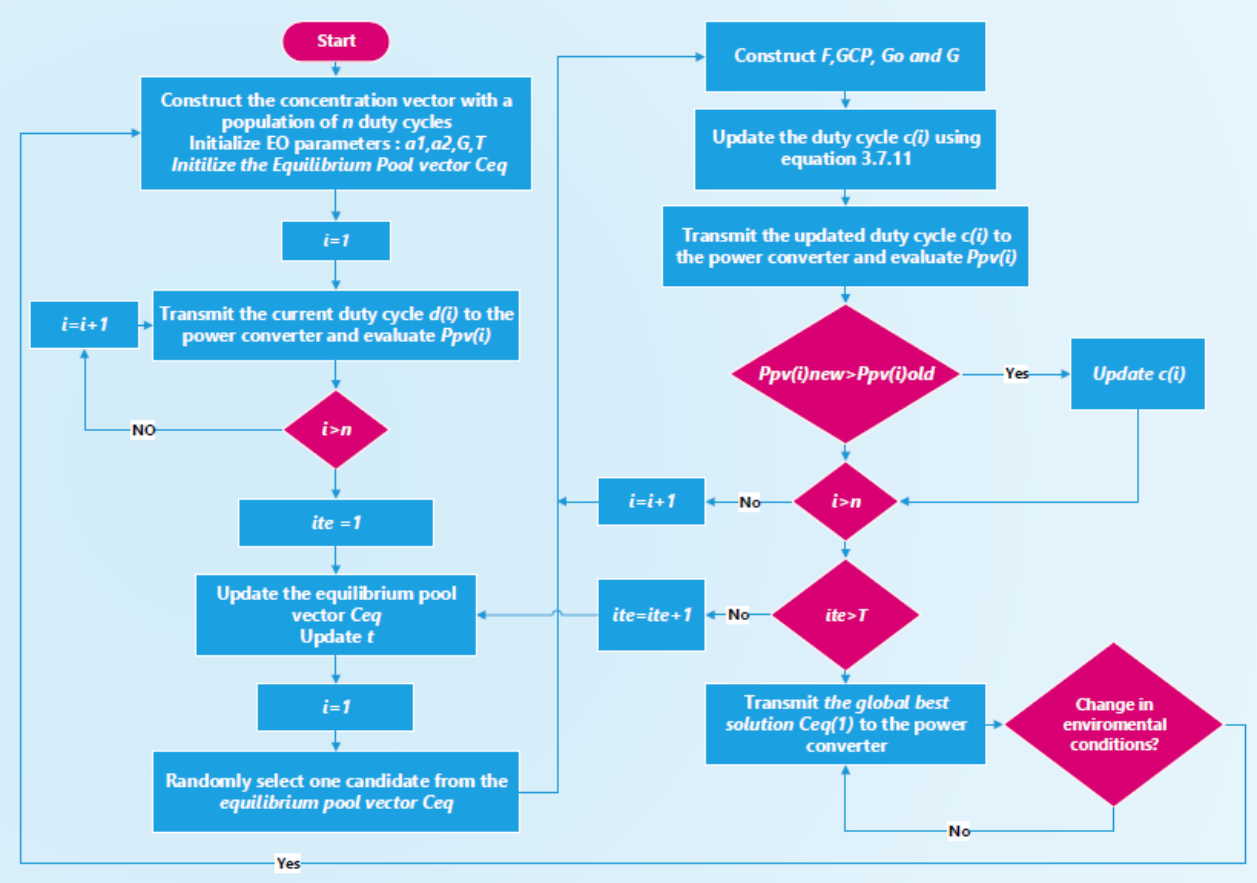

Figure 39: EO based MPPT

3.8 Seagull Optimization Algorithm:

3.8.1 Inspiration:

The Seagull Optimization algorithm is a novel bio-inspired technique for solving computationally expensive problems. The main inspiration of this algorithm is the migration and attacking strategies of a family of clever birds known as seagulls. They learn, remember and even pass on behaviors, such as stamping their feet in a group to imitate rainfall and trick earthworms to come to the surface. They even have a complex and highly developed repertoire for communication which includes a range of vocalizations and body movements [31].

Generally, seagulls live in colonies. They use their intelligence to find and attack the prey. The most important thing about seagulls is their migrating and attacking mechanisms. The following represents the key behaviors used by the Seagull Optimization technique [32].

- During migration, they travel in a group. The initial positions of seagulls are different to avoid the collisions between each other.
- In a group, seagulls can travel towards the direction of best survival fittest seagull.
- Based on the fittest seagull, other seagulls can update their initial positions.
- Seagulls frequently attack migrating birds over the sea [33] when they migrate from one place to another. They can make their spiral natural shape movement during attacking. A conceptual model of these behaviors is illustrated in Figure 40.

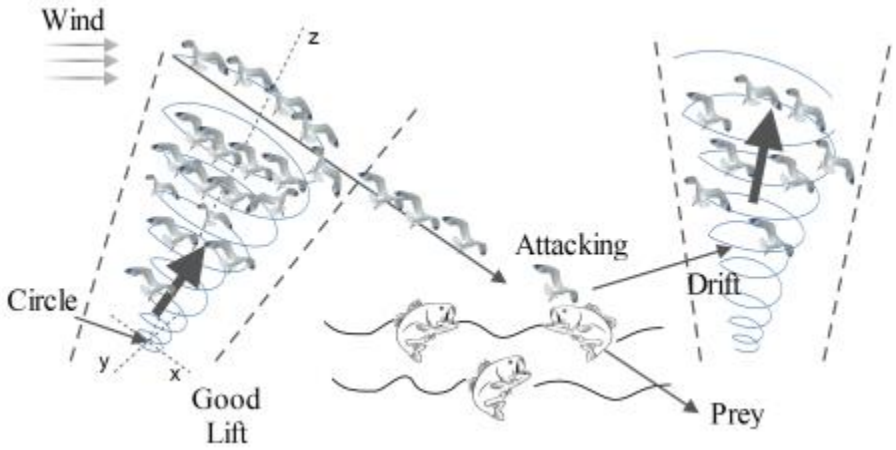

Figure 40 : Migration and Attacking Behaviors of Seagulls.

3.8.2 SOA Modeling and process steps:

A- Migration:

This stage is served as the exploration phase. The algorithm here simulates how a group of seagulls moves from one location to another. The bird should satisfy the following three conditions:

- **Avoiding Collisions**: A variable is employed to calculate the new search agent position as follows:

$$\overrightarrow{C_s} = A \times \overrightarrow{P_s(t)} \qquad 3.8.1$$

Where $\overrightarrow{C_s}$ represents the position of the search agent that does not collide with other individuals. $\overrightarrow{P_s(t)}$ is the search agent current position at iteration t, and A represents the movement behavior of the individual, and it is decreased linearly over the optimization process as follows:

$$A = f_c - f_c \times \frac{t}{T} \qquad 3.8.2$$

Where f_c is a control parameter that depends on the optimization problem.

- **Movement towards best neighbor's direction:**

After avoiding collisions between neighbors, the search agents are moved towards the direction of the best location as follows:

$$\overrightarrow{M_s} = B \times \left(\overrightarrow{P_{bs}}(t) - \overrightarrow{P_s}(t)\right) \quad 3.8.3$$

Where $\overrightarrow{M_s}$ is the individual's position towards the best search agent location, B is given in equation 3.8.4, and it is responsible for proper balancing between exploration and exploitation:

$$B = 2 \times A^2 \times rnd \quad 3.8.4$$

Where rnd is random number within the range [0,1]

- **Remain close to the best search agent:**

Lastly, the search agents will update their positions with respect to the best location using equation 3.8.5:

$$\overrightarrow{D_s} = \left|\overrightarrow{C_s} + \overrightarrow{M_s}\right| \quad 3.8.5$$

B- Migration:

The attacking process is served as the exploitation phase. Here a spiral equation is employed to mimic the helix-shaped movement of seagulls while hunting and entrapping the prey. This behavior is depicted in figure 41 and it is modeled in x, y and z planes as follows:

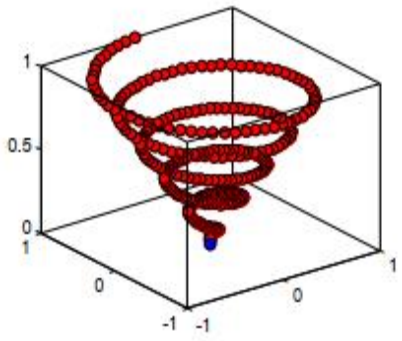

Figure 41: Natural Attacking behavior of Seagulls.

$$x' = r \times \cos(k) \quad 3.8.6$$

$$y' = r \times \sin(k) \quad 3.8.7$$

$$z' = r \times k \quad 3.8.8$$

$$r = u \times e^{kv} \qquad 3.8.9$$

Where r represents the radius of each turn of the spiral, k is a random number in the range $[0,2\pi]$, u and v are constants to define the spiral shape.

The search agents' positions are then updated using equation 3.8.10, which represents the general framework of the Seagull Optimization algorithm:

$$\overrightarrow{P_s} = \left(\overrightarrow{D_S} \times x' \times y' \times z'\right) + \overrightarrow{P_{bs}} \qquad 3.8.10$$

Where $\overrightarrow{P_s}$ saves the best individuals' positions.

3.8.3 SOA based MPPT:

The flowchart illustrating the SOA based MPP tracker is depicted in figure 42:

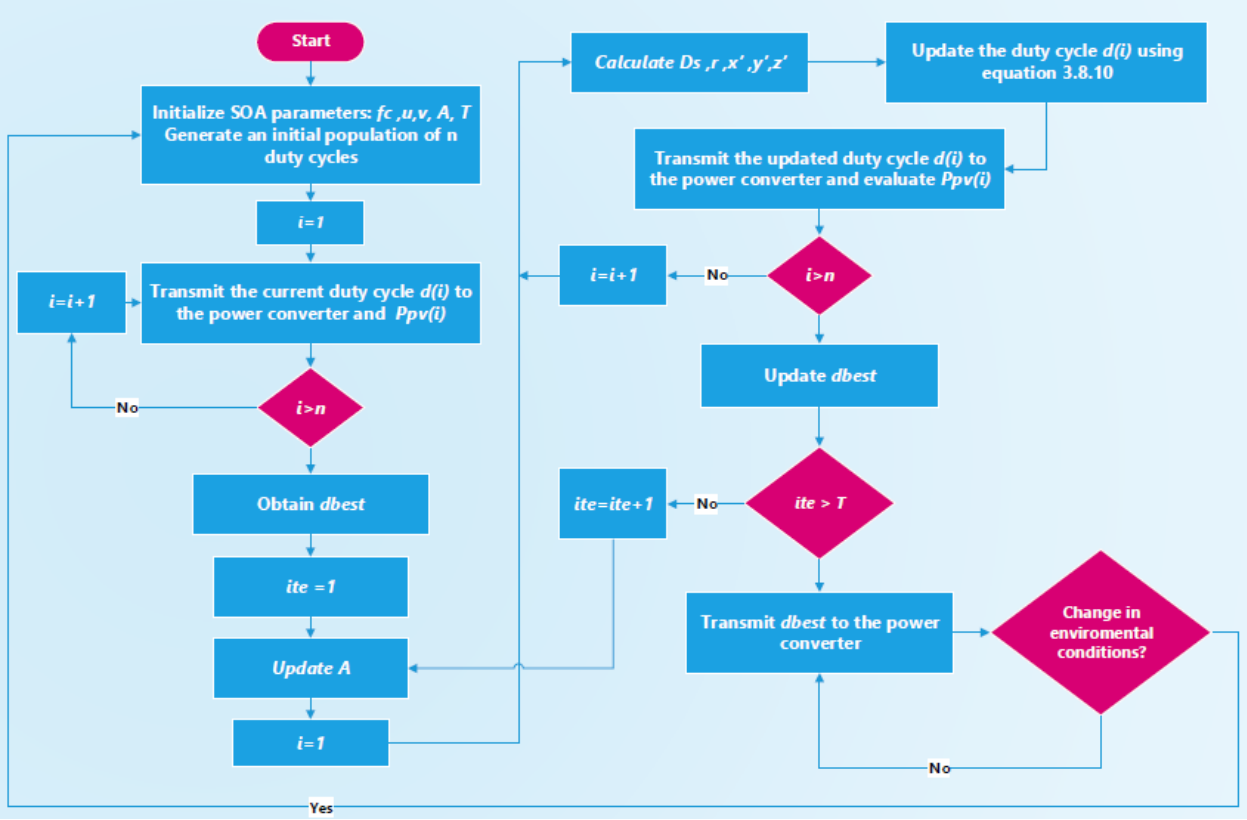

Figure 42: SOA based MPPT

3.9 Slime Mould Algorithm:

3.9.1 Inspiration:

The Slime Mould Algorithm is a novel stochastic optimizer recently proposed in 2020 [36]. The main inspiration of SMA is the peculiar foraging strategy of a type of eukaryote that lives in cool and humid places known as Slime Mould. Its main nutritional stage is Plasmodium which is the dynamic and active phase. During this focal stage, the organic matter in slime mould scouts food, surrounds it, and secretes enzymes to digest it. During the migration process, the front end extends into a fan-shaped, followed by an interconnected venous network that allows cytoplasm to drift inside. Because of their unique pattern and characteristic, they can use multiple food sources at the same time to form a venous network connecting them as shown in figure 43.

In [34], a detailed study of the cytoplasmic flow of Slime mould helped the developers of the SMA to lay down its foundations. When a vein approaches a food source, the bio-oscillator produces a propagating wave [35] that increases the cytoplasmic flow through the vein, and the faster the cytoplasm flows, the thicker the vein. Through this combination of positive-negative feedback, the slime can establish the optimal path to connect food in a relatively superior way.

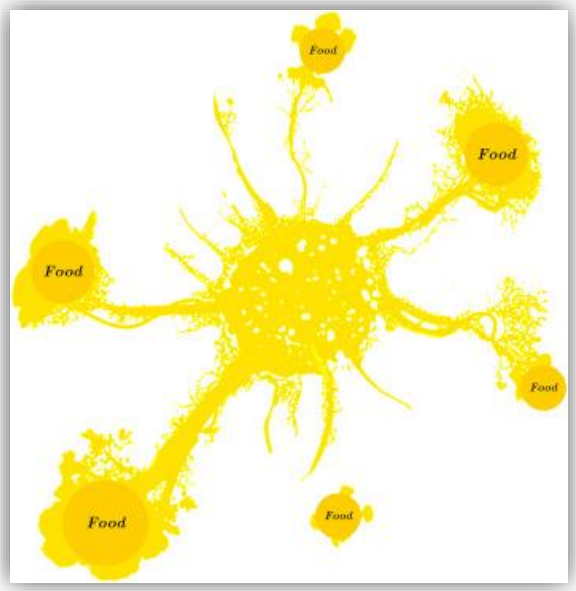

Figure 43: Foraging Morphology of Slime Mould

3.9.2 Mathematical Modeling and Process Steps:

A- Approach food:

Slime mould can approach food according to the odor in the air. To express its approaching behavior in mathematical formulas, the following formulas are proposed to imitate the contraction mode:

$$\overrightarrow{X(t+1)} = \begin{cases} \overrightarrow{X_b(t)} + \overrightarrow{vb} \cdot \left(\overrightarrow{W} \cdot \overrightarrow{X_A(t)} - \overrightarrow{X_B(t)} \right), r < p \\ \overrightarrow{vc} \cdot \overrightarrow{X(t)}, r \geq p \end{cases} \quad 3.9.1$$

Where $\overrightarrow{vb}$ is a parameter with a range of [-a, a], $\overrightarrow{vc}$ decreases linearly from one to zero. t represents the current iteration, $\overrightarrow{X_B}$ represents the individual location with the highest odor concentration currently found, $\overrightarrow{X}$ denotes the location of slime mould, $\overrightarrow{X_A}$ and $\overrightarrow{X_B}$ are two individuals randomly selected from the population, and $\overrightarrow{W}$ represents the weight of slime mould. p is given as follows:

$$p = tanh|S(i) - DF| \quad 3.9.2$$

where $i \in 1, 2, \ldots, n$, $S(i)$represents the fitness of $\overrightarrow{X}$,and DF represents the best fitness obtained so far.

$\overrightarrow{vb}$, a and $\overrightarrow{W}$ are given as follows:

$$\overrightarrow{vb} = [-a, a] \quad 3.9.3$$

$$a = \text{arctanh}(-\left(\frac{t}{\text{T}}\right) + 1) \quad 3.9.4$$

$$\overrightarrow{W(SmellIndex(i))} = \begin{cases} 1 + r \cdot log\left(\frac{bF - S(i)}{bF - wF} + 1\right), condition \\ 1 - r \cdot log\left(\frac{bF - S(i)}{bF - wF} + 1\right), others \end{cases} \quad 3.9.5$$

Where, condition indicates that $S(i)$ ranks first half of the population, r is a random value in the interval of [0, 1], bF denotes the optimal fitness obtained in the current iterative process, wF denotes the worst fitness value obtained in the iterative process currently, $SmellIndex$ denotes the sequence of fitness values sorted.

$$SmellIndex = sort(S) \quad 3.9.6$$

B- Wrap food:

When the food concentration is content, the bigger the weight near the region is; when the food concentration is low, the weight of the region will be reduced, thus turning to explore other regions. Based on the above principle, the mathematical formula for updating the location of slime mould is as follows [36]:

$$\overrightarrow{X\,(t)} = \begin{cases} rand\,.(UB - LB) + LB, & rand < z \\ \overrightarrow{X_b(t)} + \overrightarrow{vb}\,.\left(W.\overrightarrow{X_A(t)} - \overrightarrow{X_B(t)}\right), & r < p \\ \overrightarrow{vc}\,.\overrightarrow{X\,(t)}, & r \geq p \end{cases} \quad 3.9.7$$

where LB and UB denote the lower and upper boundaries of the search range, rand is a random value within [0,1]. The value of z is discussed in the parameter setting section of chapter 4.

C- Grabble food:

On the purpose of simulating the variations of venous width of slime mould, W, vb and vc are used to realize the variations.W mathematically simulates the oscillation frequency of slime mould near one at different food concentration, so that slime mould can approach food more quickly when they find high-quality food, while approach food more slowly when the food concentration is lower in individual position, thus improving the efficiency of slime mould in choosing the optimal food source.vb fluctuates randomly within the interval [-a, a], while vc oscillates between [-1,1] and both will approach zero eventually.

The process steps of the SMA can be summarized as follows:
Initialize the parameters: *population size* n, *Maximum number of iterations T*
Initialize the positions of slime mould $X_i(i = 1,2, \dots, n)$
While $(ite \leq T)$
Calculate the fitness of all slime mould;
Update bestFitness X_b;
Calculate the weight W using equation 3.9.5;
For each slime mould position:
Update p, vb, vc;
Update positions using equation 3.9.7;
End For
$ite = ite + 1$;
End While
Return X_b

3.9.3 SMA based MPPT

The implementation of the SMA in MPPT is illustrated in the flowchart of figure 44.

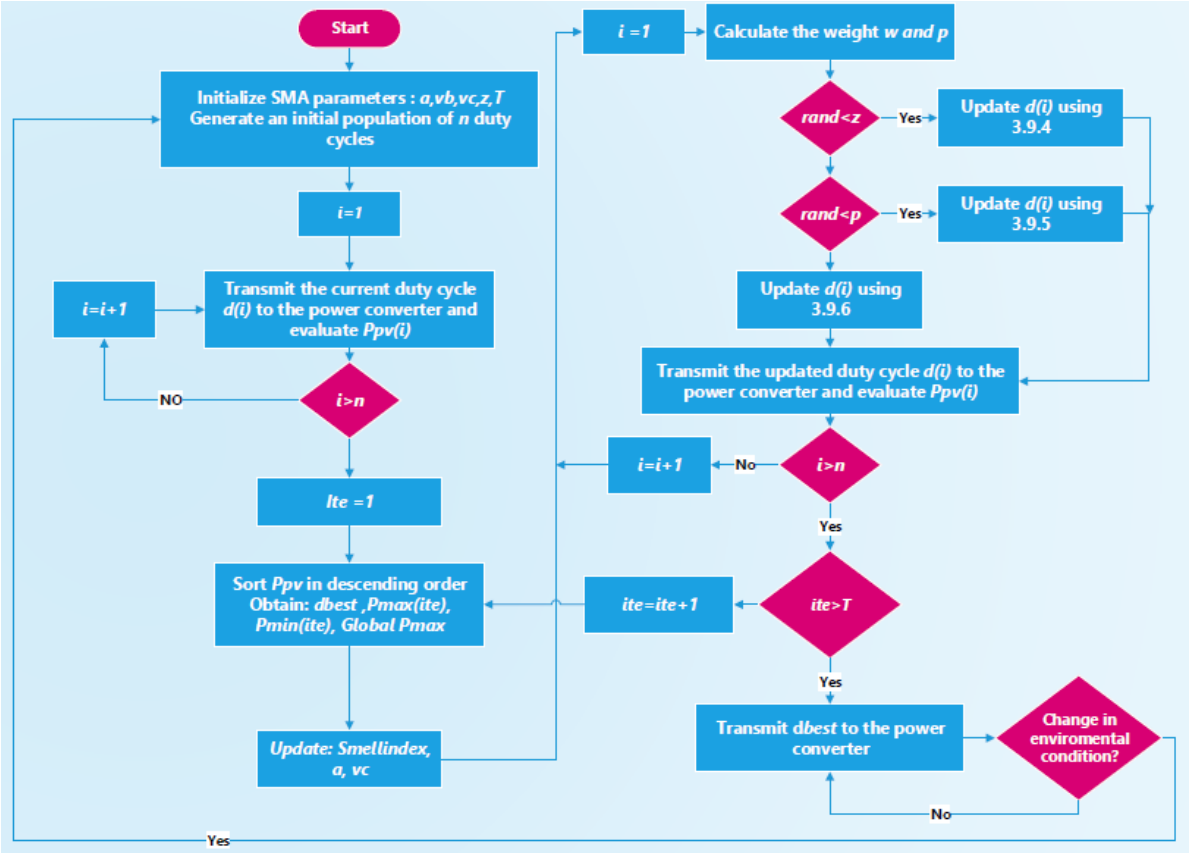

Figure 44: SMA based MPPT

3.10 Conclusion:

This chapter set forth five well-known nature inspired algorithms, as well as the latest developments in the field of metaheuristics. A survey that consists of the inspiration and mathematical modelling of these optimizers, and their operating flowcharts in maximum power tracking, was provided. Moreover, we have also laid out three novel algorithms that are recently proposed, namely: Equilibrium Optimizer, Seagull Optimization and the Slime Mould Algorithm. In the next chapter, we are going to evaluate the propound techniques in an MPPT based standalone PV system.

CHAPTER 4 : Simulation and Results

4.1 Introduction:

To assess the effectiveness of the proposed techniques, Matlab and Simulink are used to simulate a simple stand-alone system subjected to several atmospheric conditions. A comparative study in terms of robustness, tracking speed and efficiency is provided at the end of this chapter.

4.2 System Overview:

The Simulink model of our system is illustrated in figure 45. It consists essentially of two serially connected PV modules of the same type, a boost converter driven by an MPPT controller, and a load of 40Ω. Usually, a standard 60 cell PV module is built from 3 substrings, each is protected by a bypass diode [16]. The 3 substrings are serially connected to each other to form a single module.

4.3 DC-DC boost converter design:

Using the results of section 3.4, and setting the switching frequency to 50 KHz , the components of the power converter are selected as shown in table 2:

Table 2: DC-DC Converter Components Selection for System 2

Switching Frequency	**Inductor (L)**	**Input Capacitor**	**Output Capacitor**
50 KHz	4 μH	100μF	1μF

It is worth mentioning that in order to get accurate measurements of the output PV power that corresponds to each duty cycle, the time interval between two successive transmissions of D should be greater than the boost converter settling time. To do so, several values of the duty cycle have to be tested to analyze the transient response of the power converter and the time it takes to settle down. After performing this evaluation, we have found that a sampling time of 0.008s is appropriate for our system.

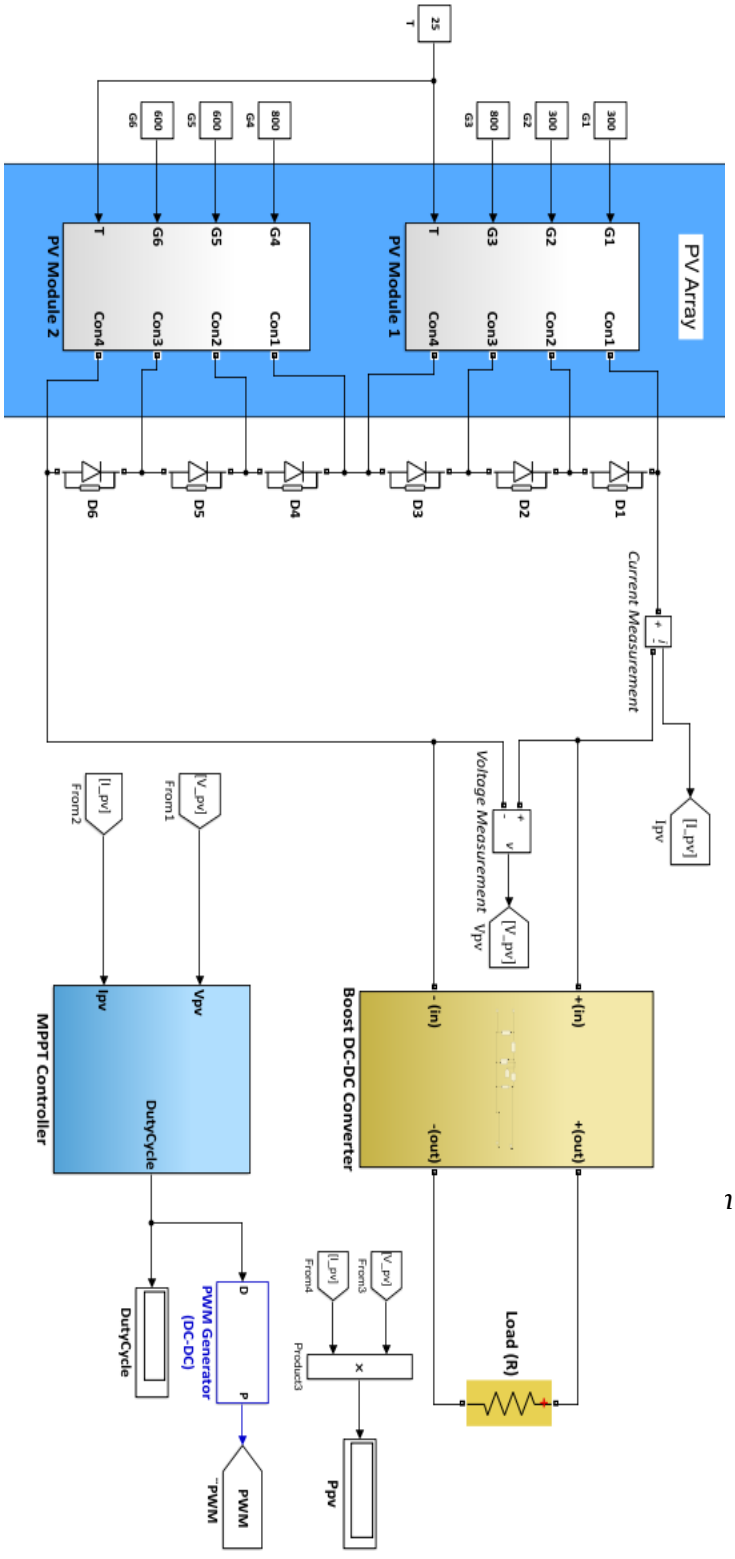

Figure 46: Simulink Model of the designed System -2-

4.4 Algorithms Parameterization:

Table 3 provides the selected parameters of every optimizer, the reader should notice that the settings may differ to that provided in the original papers of these algorithms that might be unsuitable for MPPT applications.

The selection of the population size N has a significant impact on the performance of the optimizer. A large N will improve the search ability and tracking accuracy; however, it will increase the convergence time of the algorithm. On the other hand, a small N speeds up the search process, but it may lead to poor power efficiency.

In this work, and in order to make a fair comparison, we have selected N to be 4, which seems to be a reasonable choice. The maximum number of iterations is set to 10 for all algorithms.

Table 3: Algorithms Parametrization

Algorithms	***Parametrization***
PSO	$\omega = 0.4 c_1 = 1.4 \quad c_2 = 2$
WOA	$a = 1 - t/T$ b=1
GWO	$a = 0.7 - 0.7t/T$
WDO	$g = 0.5 - 0.5t/T$ $V_{Max} = 0.3 - 0.3t/T$ $c = 0.5 - 0.5t/T$ $RT = 3 \quad a = 0.8$
GOA	$c = 0.1 - 0.004t/T$
EO	$V = 1\,, a_1 = 1.6\,, a_2 = 1, GP = 0.5;$
SOA	$A = 0.7 - 0.7t/T$ $u = 0.8\,, v = 0.05$
SMA	$a = \tan^{-1}(0.85 - \frac{0.85t}{T})$ $b = 1.2 - 0.64t/T$ $z = 0.03$

4.5 Results and Discussion:

Six distinct scenarios have been subjected to the PV system. In the first case, the PV array receives a fast-varying uniform irradiance that changes every 0.5 seconds in three different levels: 500-1000-750 *W/m²*. In the five remaining cases, different partial shading patterns with various numbers of peaks have been exposed to the substrings that constitute the two PV modules. Table 4 and figure 46 illustrate the different scenarios and their PV characteristics.

Table 4 : Studied Irradiance conditions

Cases	***Irradiance levels distribution on the modules Substrings (W/m²)***	***GMPP (W)***
1: Uniform Fast Varying Irradiance	[0s,0.5s]:500 [0.5s,1s]:1000 [1s,1.5s]:750	216.2 426.3 322.9
2: PSC	300/300/800/800/600/600	183.196
3: PSC	1000/1000/600/300/300/800	192.774
4: PSC	1000/400/800/800/300/600	191.184
5: PSC	1000/500/800/700/300/600	201.93
6: PSC	1000/500/800/700/900/600	245.1681

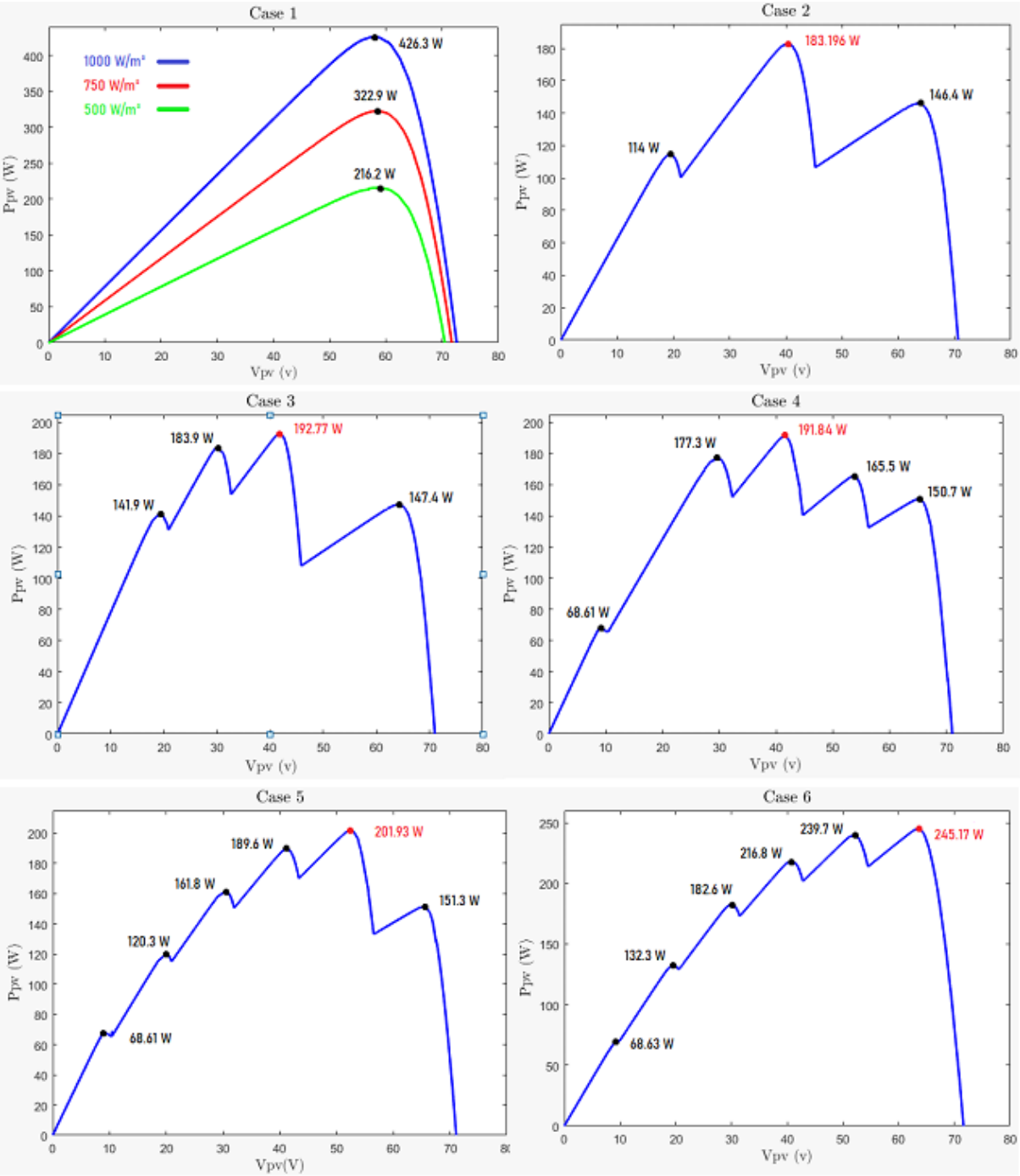

Figure 47: P-V Characteristic Curves of the Studied Cases

4.5.1 Uniform Fast Varying irradiance:

The resulting PV power curves for each algorithm with uniform fast varying irradiation are shown in figure 47. All algorithms were able to detect changes in irradiation levels, and locate the maximum power point. Table 5 provides the results details of the simulation in terms of efficiency and time convergence in each time interval

Table 5: Steady State Tracking Results Under Fast Varying Uniform Irradiance

Optimizer	GMPP(W)	Tracked Power (W)	Convergence Time (s)
PSO	216.2 426.3 322.9	216.15-426.19-322.89	0.29-0.25-0.24
WOA		216.15-426.16-322.87	0.27-0.27-0.27
GWO		216.15-425.94-322.89	0.265-0.295-0.28
WDO		216.14-426.03-322.9	0.24-0.24-0.24
GOA		216.15-426.185-322.888	0.24-0.28-0.278
EO		216.15-426.18-322.9	0.22-0.28-0.27
SOA		216.14-426.19-322.9	0.26-0.26-0.26
SMA		216.15-426.19-322.9	0.265-0.29-0.29

In order to have a better assessment measure, we consider the average efficiencies and average convergence time as provided in table

Table 6: Average Efficiencies and Convergence Time Under Fast Varying Uniform Irradiation

Optimizer	Average Efficiency (%)	Average Convergence Time (s)
PSO	99.98	0.26
WOA	99.97	0.27
GWO	99.96	0.28
WDO	99.97	0.248
GOA	99.98	0.24
EO	99.98	0.256
SOA	99.98	0.26
SMA	99.98	0.28

It can be observed, that all algorithms have nearly equal efficiencies, but with a considerable difference in convergence time. With GOA, WDO and EO being fastest respectively.

4.5.2 Non-uniform Irradiance levels:

Figures 48, 49, 50, 51, 52 depict the obtained power curves under partial shading conditions in cases 2 through 6, and table 7 provides the resulting steady state static efficiencies and tracking times.

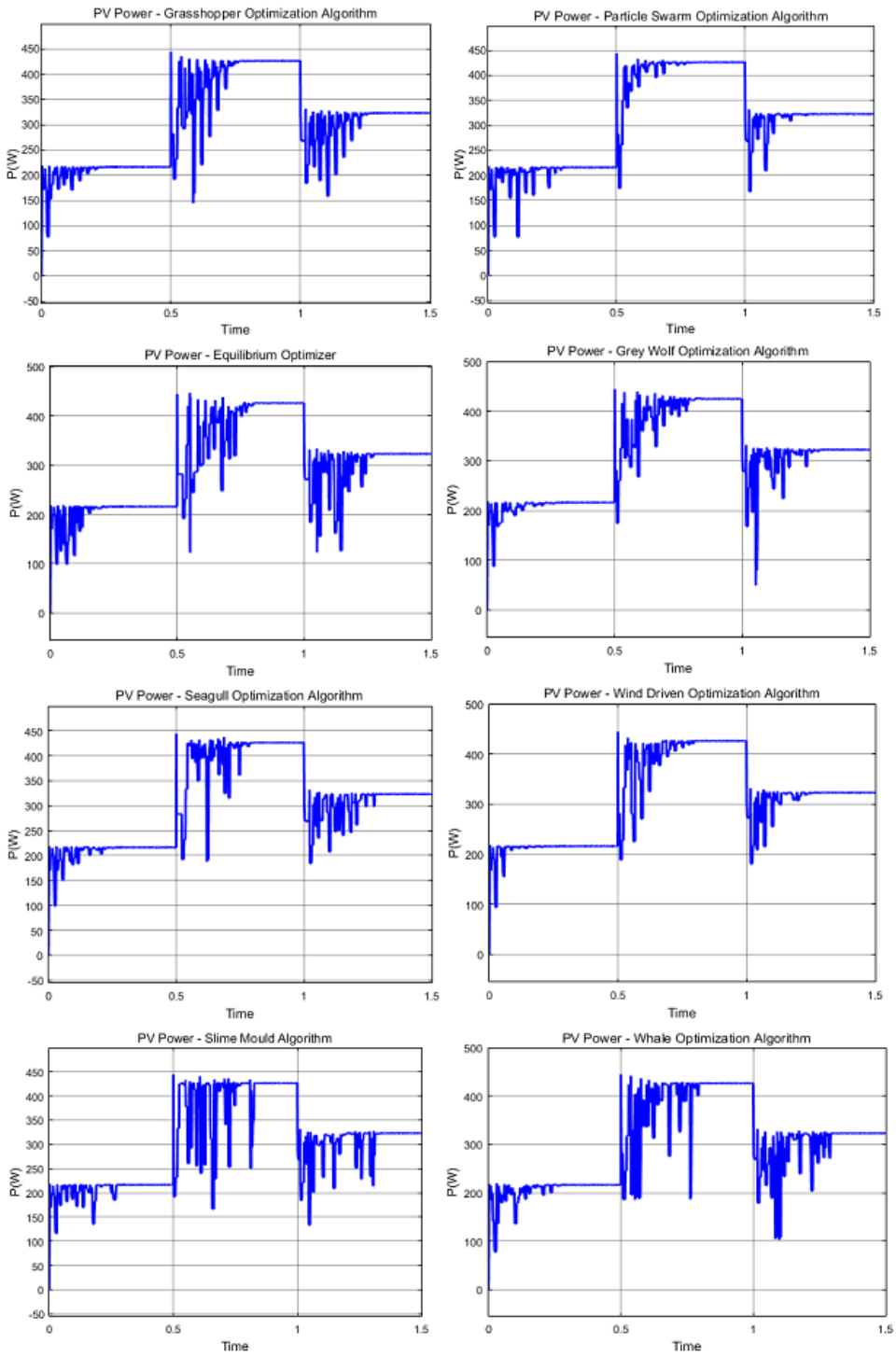

Figure 48: PV Power Curves of Case 1

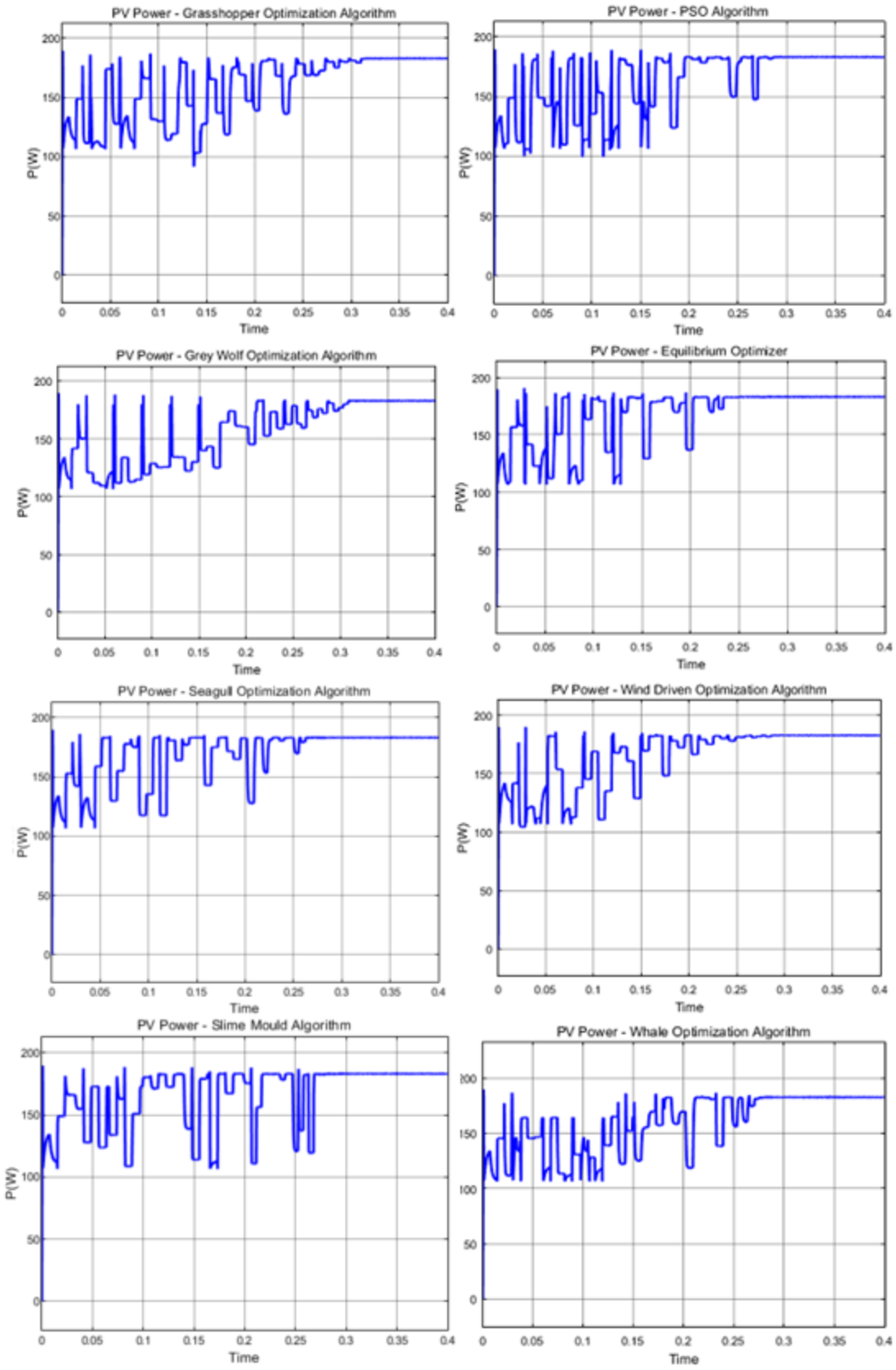

Figure 49: PV Power Curves of Case 2

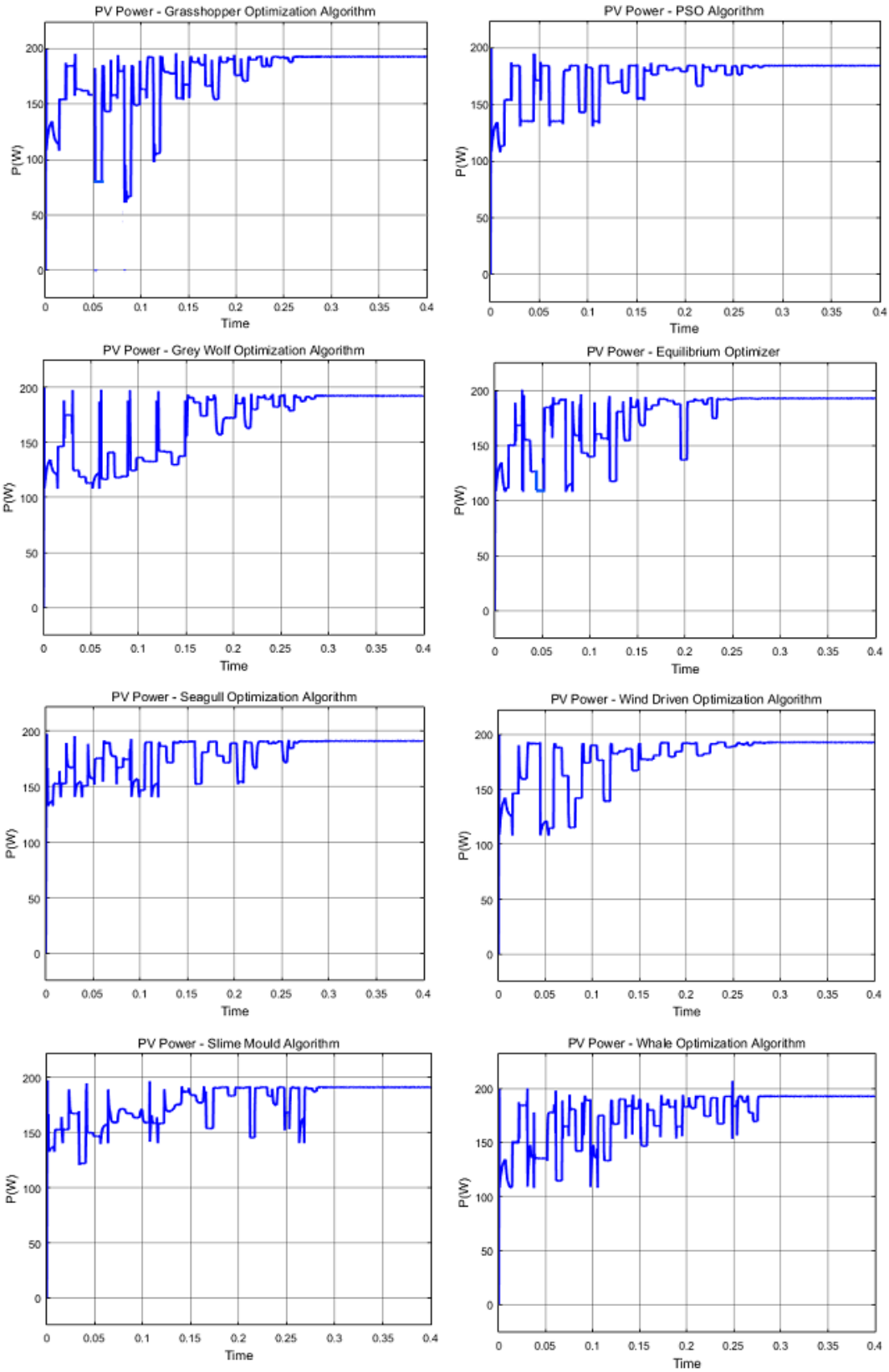

Figure 50: PV Power Curves of Case 3

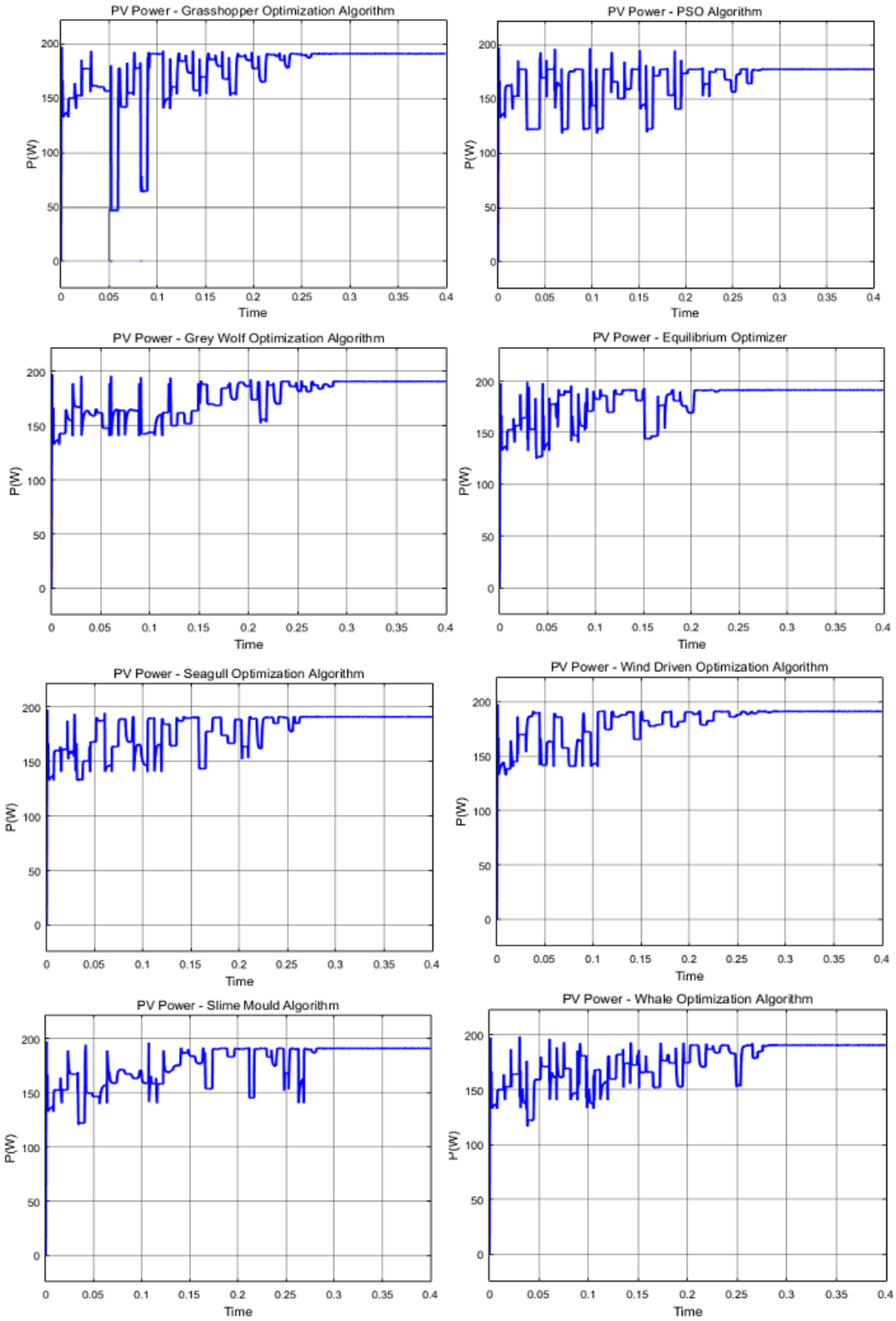

Figure 51:PV Power Curves of Case 4

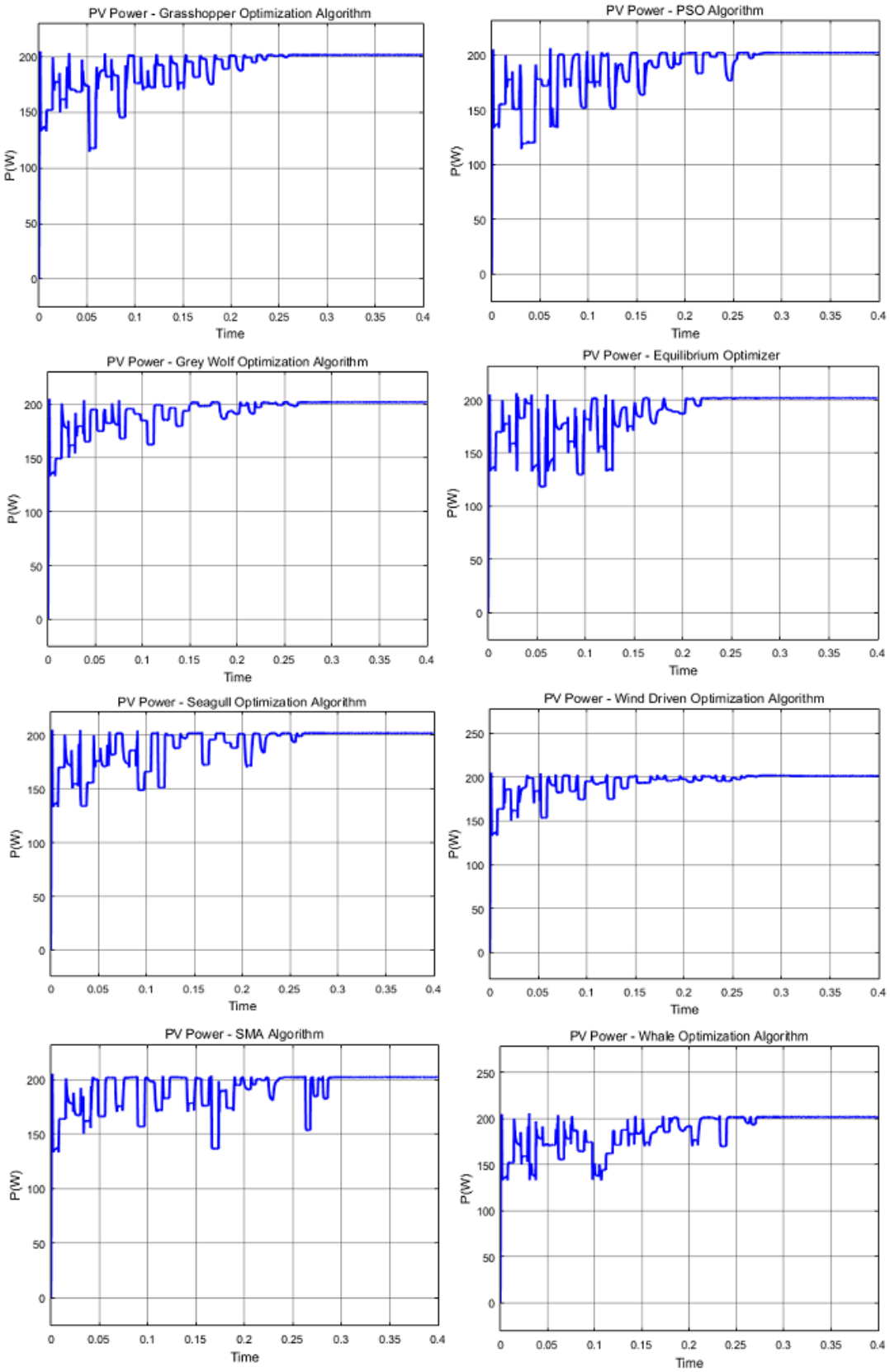

Figure 52:PV Power Curves of Case 5

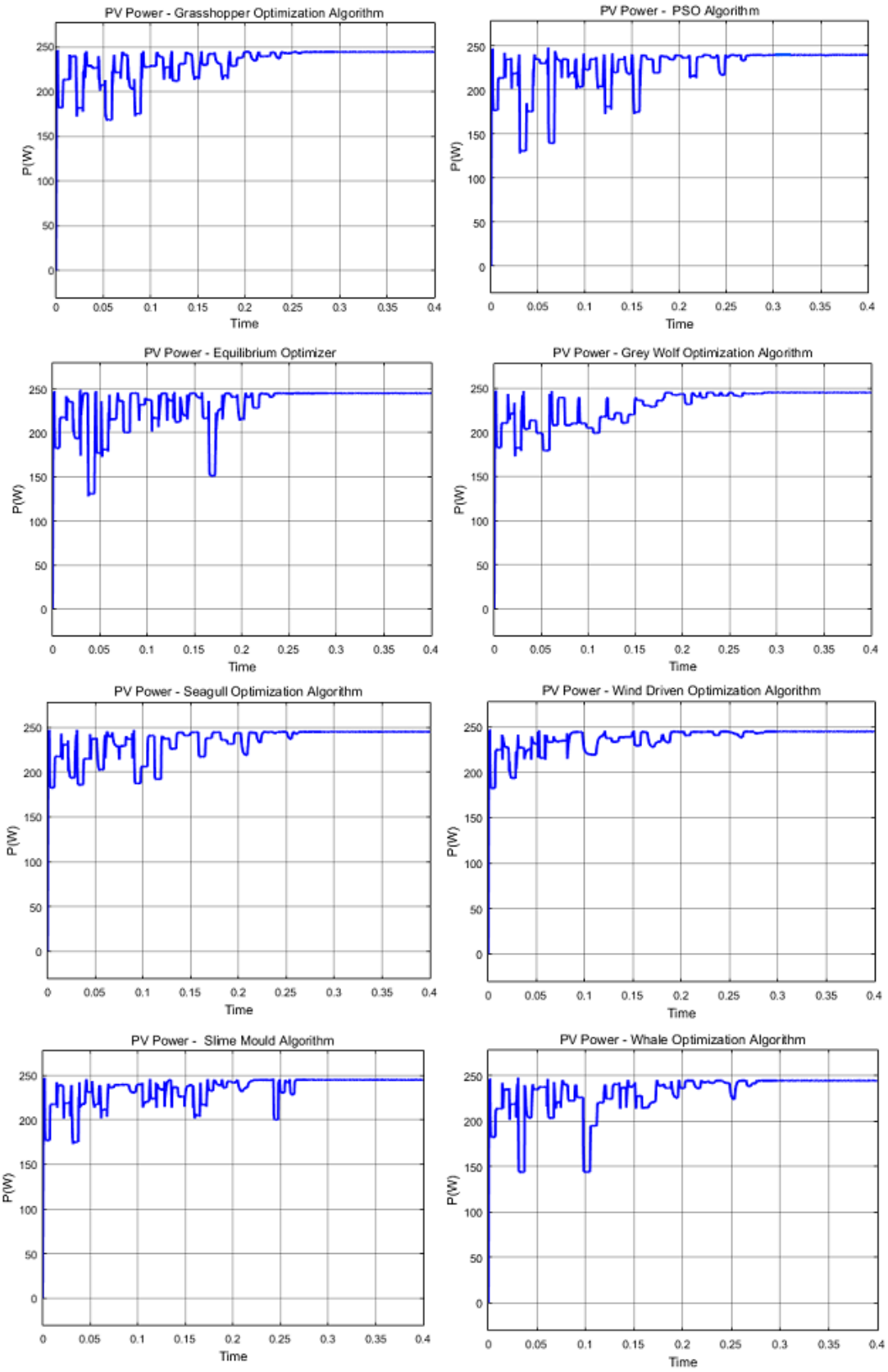

Figure 53:PV Power Curves of Case 6

Table 7: Steady State Tracking Results Under Non-Uniform Irradiance

Cases	Optimizer	GMPP(W)	Tracked Power (W)	GMPP Located	Efficiency (%)	Convergence Time (s)
2	PSO	183.196	182.98	Yes	99.88	0.285
	WOA		182.49	Yes	99.61	0.27
	GWO		182.86	Yes	99.81	0.29
	WDO		182.54	Yes	99.64	0.27
	GOA		182.98	Yes	99.88	0.26
	EO		182.98	Yes	99.88	0.235
	SOA		182.98	Yes	99.88	0.26
	SMA		182.98	Yes	99.88	0.268
3	PSO	192.774	177.5	**No**	92.07	0.26
	WOA		192.6	Yes	99.91	0.275
	GWO		192.15	Yes	99.67	0.29
	WDO		192.66	Yes	99.94	0.27
	GOA		192.66	Yes	99.94	0.26
	EO		192.66	Yes	99.94	0.24
	SOA		192.673	Yes	99.947	0.26
	SMA		192.68	Yes	99.95	0.285
4	PSO	191.184	191.1	Yes	99.95	0.285
	WOA		190.54	Yes	99.66	0.28
	GWO		190.95	Yes	99.88	0.285
	WDO		191.1	Yes	99.95	0.275
	GOA		191.11	Yes	99.96	0.26
	EO		191.11	Yes	99.96	0.23
	SOA		191.1	Yes	99.95	0.26
	SMA		191.1	Yes	99.95	0.267
5	PSO	201.93	201.84	Yes	99.955	0.29
	WOA		201.83	Yes	99.95	0.27
	GWO		201.83	Yes	99.95	0.27
	WDO		201.81	Yes	99.94	0.275
	GOA		201.85	Yes	99.96	0.26
	EO		201.84	Yes	99.955	0.22
	SOA		201.845	Yes	99.957	0.26

	SMA		201.85	Yes	99.96	0.285
6	PSO	245.1681	239.82	**No**		0.29
	WOA		244.63	Yes	99.78	0.27
	GWO		244.73	Yes	99.82	0.28
	WDO		244.75	Yes	99.83	0.28
	GOA		244.75	Yes	99.83	0.25
	EO		244.76	Yes	99.83	0.23
	SOA		244.76	Yes	99.83	0.26
	SMA		244.75	Yes	99.83	0.265

- It can be observed that the EO, SOA, SMA and the GOA scored almost equal efficiencies, and successfully located the GMPP associated with each case. The WDO have also achieved Power levels as high as the other techniques, however in the second case, it obtained the second worst efficiency (99.64%), compared to the best one (99.94%).
- The WOA got the lowest power level in the second case at which it achieved 182.49W out of 183.196W resulting in an efficiency of 99.61% compared to the highest rank (99.88%). And also in the 4th case at which it hit 190.95W out of 191.184W resulting in an efficiency of 99.877%. In the remaining shading conditions, the WOA attained efficiency levels equal or near to the highest ones.
- The worst result obtained by the GWO algorithm was in 3rd case at which it obtained an efficiency 99.67% which can be ranked at the 6th position in descending order.
- The PSO algorithm in the other hand have stagnated at the first local maximum associated with the 6th and 3rd shading patterns (239.82W out of 245.168 in case 6 and 177.5 W out of 192.774W in case 3).
- In terms of convergence speed, it can be clearly seen that the Equilibrium optimizer is the fastest among the remaining algorithms with settling time as low as 0.22s in the 5th case up to 0.24s in the 3rd case. This is due its framework equation, which is based on updating the population using the best-so-far-solutions. Using this strategy, the algorithm has high prospects of exploring promising regions rapidly, and hence undesirable areas are replaced along the optimization process, resulting in an efficacious transition from diversification into intensification.
- The SOA can be classified in the second rank with a convergence time of 0.26 seconds in all cases, which is identical to the Grasshopper Optimization algorithm.
- The Slime Mould Algorithm in the other hand, can be put in the third rank with convergence time of 0.265s in the sixth case which is its best score, and 0.285s in cases 3 and 5. The latter outcome is the direct result of the first part in equation 3.10.7, which occasionally drives a certain agent into a random position when the stated condition ($rand() < z$) is satisfied, at any iteration throughout the optimization process.

Table 8 provides the average efficiencies and average settling times obtained by the assessed techniques:

Table 8: Average Efficiencies and Convergence Time Under Non-Uniform Irradiation

Optimizer	Average Efficiency (%)	Average Convergence Time(s)
PSO	98.32	0.28
WOA	99.78	0.273
GWO	99.826	0.283
WDO	99.86	0.274
GOA	99.912	0.26
EO	**99.913**	**0.23**
SOA	99.913	0.26
SMA	99.914	0.274

4.6 Conclusion:

This chapter was devoted to the evaluation of 8 metaheuristic algorithms in Maximum Power Point Tracking. Simulink and Matlab were used to design and simulate a standalone PV system driven by an MPPT controller and subjected to various atmospheric conditions. The assessment was carried out based on the collected data obtained from 6 distinct scenarios of fast varying uniform and non-uniform irradiation, accompanied with the resulting power curves and necessary tables. The analyses of the simulation results of the studied cases, have demonstrated the effectiveness of the propound novel algorithms in handling various challenging shading patterns, and achieved the highest efficiency levels in all cases over the remaining popular stochastic algorithms. Moreover, the three proposed optimizers are characterized by fast tracking speed, this was conspicuous in the convergence time of the Equilibrium optimizer which was 0.23s on average outperforming the remaining algorithms, and the Seagull Optimization Algorithm which was a powerful competitor to the well-known Grasshopper Optimization Algorithm with nearly identical settling times of 0.26 seconds in most of the studied cases.

General Conclusion and Future Work

This master thesis work was devoted to the study and evaluation of the latest advancements of soft computing techniques for Maximum Power Point Tracking operation. Equilibrium Optimizer, Seagull Optimization and the Slime Mould Algorithm are the three recent optimizers investigated in this project and assessed along with other well-known metaheuristic techniques, namely: Particle Swarm Optimization, Whale Optimization, Grey Wolf Optimization, Wind Driven Optimization and the Grasshopper Optimization. The proposed algorithms have proved to be powerful in terms of tracking efficiency, robustness and convergence time. The comparative study revealed the fast-tracking outperformance of the Equilibrium Optimizer over the remaining techniques, and the nearly identical outcomes obtained by the SOA and the GOA. It is worth highlighting that although the SMA was on a par with the other optimizers in terms of efficiency, some improvements need to be made, to make it faster and a better rival.

Although the propound metaheuristic techniques are powerful for Maximum Power Point Tracking, the work can be further extended, and enhancements can always be made in this field. In this context, we suggest that the following points are worth investigating for future work:

- A hybridization of the assessed algorithms with classical techniques can be made to reduce the convergence time of the MPPT controller. This allows exchanging between local tracking and global tracking strategies according to the solar irradiance conditions. As global maximum power point tracking takes longer time, and if the solar irradiance is uniform, it is needless to employ a global tracker, since a traditional technique like the P&O is sufficient and faster. In the other hand, if partial shading occurs, then the MPPT controller will be switched to global search using one of the proposed metaheuristic algorithms. A partial shading detection technique has to be developed for that purpose.
- Evaluate the proposed algorithms under varying load and temperature conditions.
- Build a laboratory setup based on DSP or FPGA boards to implement and experimentally validate the feasibility of the assessed techniques in Maximum Power Point Tracking.

References:

[1] https://www.altestore.com/howto/components-for-your-solar-panel-photovoltaic-system-a82/. Accessed August 2020.

[2] Bo Zhao, Caisheng, Wang Xuesong, Zhang (2017). *Grid-Integrated and Standalone Photovoltaic Distributed Generation Systems: Analysis, Design, and Control*. Springer International Publishing.

[3] https://www.youtube.com/watch?v=MCd9Fxl7ZM&list=PL2fCZiDqOYYXb8Q5HBXnU_-87WDuNZdW1&index=52&ab_channel=edX. Accessed August 2020.

[4] https://circuitglobe.com/photovoltaic-or-solar-cell.html. Accessed July 2020.

[5] S. Sumathi,L. Ashok Kumar,P. Surekha. (2015). *Solar PV and Wind Energy Conversion Systems: An Introduction to Theory, Modelling with MATLAB/SIMULINK, and the Role of Soft Computing Techniques*. Springer International Publishing.

[6] Nur Atharah Kamarzaman, Chee Wei Tan. (2014). A comprehensive review of maximum power point tracking algorithms for photovoltaic systems. *Renewable & Sustainable Energy Reviews*, 37, 585-598

[7] Yiwang Wang, Yong Yang, Gang Fang, Bo Zhang, Huiqing Wen, Houjun Tang, Li Fu and Xiaogao Chen. (2018). An Advanced Maximum Power Point Tracking Method for Photovoltaic Systems by Using Variable Universe Fuzzy Logic Control considering Temperature Variability. *electronics*, *7*(12), 355.

[8] D. Yogi Goswami, Frank Kreith and Jan F. Kreider. (2000). Principles of Solar Engineering, Taylor & Francis, USA, 2000.

[9] https://www.energy.gov/eere/solar/articles/solar-performance-and-efficiency#:~:text=The%20conversion%20efficiency%20of%20a,is%20converted%20into%20usable%20electricity. Accessed July 2020.

[10] Mohamed Ahmed Hassan El-Sayed, Steven B. Leeb. (2017). *MAXIMUM POWER POINT TRACKING OF PV ARRAYS IN KUWAIT*.

[11] Syafaruddin, Engin Karatepe, Takashi Hiyama. (2012). Performance enhancement of photovoltaic array through string and central based MPPT system under non-uniform irradiance conditions. *Energy Conversion and Management*, 62, 131-140.

[12] S. Motahhir, A. El Ghzizal, S. Sebti, and A. Derouich. (2016). Shading effect to energy withdrawn from the photovoltaic panel and implementation of DMPPT using C language. *International Review of Automatic Control*, 9(2), 88 - 94.

[13] Jaycar Electronics Reference Data Sheet: DCDCCONV.PDF

[14] Daniel Hart. (2010). *Power Electronics*. McGraw-Hill Education.

[15] Saad Motahhir, Abdelaziz El Ghzizal, Souad Sebti, and Aziz Derouich. (2018). Modeling of Photovoltaic System with Modified Incremental Conductance Algorithm for Fast Changes of Irradiance. *International Journal of Photoenergy*, 2018, 1-13.

[16] Performance of PV Topologies under Shaded Conditions, 2013, *SolarEdge.*

[17] Xin-She Yang. (2014). *Nature-Inspired Optimization Algorithms*. Elsevier.

[18] Ke-Lin Du, M. N. S. Swam. (2016). *Search and Optimization by Metaheuristics: Techniques and Algorithms Inspired by Nature*. *Birkhäuser*.

[19] Abdulkadir, M. & Yatim, Abdul Halim & Yusuf, S. (2014). An improved PSO-based MPPT control strategy for photovoltaic systems. *International Journal of Photoenergy*, 2014(7), 1-11.

[20] Mirjalili, Seyedali & Lewis, Andrew. (2016). The Whale Optimization Algorithm. *Advances in Engineering Software*, 95, 51-67.

[21] Faris, H., Aljarah, I., Al-Betar, M.A. et al. (2018). Grey wolf optimizer: a review of recent variants and applications. Neural Comput & Applic, 30, 413–435. https://doi.org/10.1007/s00521-017-3272-5

[22] Mirjalili, S. M. and Andrew P. Lewis. (2014). Grey Wolf Optimizer. *Advances in Engineering Software*. 69, 46-61.

[23] Zhang, Sen & Luo, Qifang & Zhou, Yong-Quan. (2017). Hybrid Grey Wolf Optimizer Using Elite Opposition-Based Learning Strategy and Simplex Method. *International Journal of Computational Intelligence and Applications*. 1750012. 10.1142/S1469026817500122.

[24] R. D. Thompson. (1998). Atmospheric Processes and Systems, Routledge, New York.

[25] C. D. Ahrens. (2003). Meteorology Today: An Introduction to Weather, Climate, and the Environment, 7th ed., Thomson–Brook/Cole.

[26] Z. Bayraktar, M. Komurcu, J. A. Bossard and D. H. Werner. (2013). The Wind Driven Optimization Technique and its Application in Electromagnetics. *IEEE Transactions on Antennas and Propagation*, 61, 5, 2745-2757. Doi: 10.1109/TAP.2013.2238654.

[27] Ahsan, Waleed & Khan, Muhammad & Aadil, Farhan & Maqsood, Muazzam & Ashraf, Staish & Nam, Yunyoung & Rho, Seungmin. (2020). Optimized Node Clustering in VANETs by Using Meta-Heuristic Algorithms, *Electronics*. 9(3), 394.

[28] Saremi, Shahzad & Mirjalili, Seyedali & Lewis, Andrew. (2017). Grasshopper Optimization Algorithm: Theory and application. *Advances in Engineering Software*, 105, 30-47. 10.1016/j.advengsoft.2017.01.004.

[29] W.W. Nazaroff, L. Alvarez-Cohen, Environmental Engineering Science, 1 edition (2020), Wiley, New York.

[30] Faramarzi, Afshin & Heidarinejad, Mohammad & Stephens, Brent & Mirjalili, Seyedali. (2019). Equilibrium optimizer: A novel optimization algorithm. *Knowledge-Based Systems*, 191. 10.1016/j.knosys.2019.105190.

[31] https://onekindplanet.org/animal/seagull/.

[32] Dhiman, Gaurav & Chahar, Vijay. (2018). Seagull Optimization Algorithm: Theory and its Applications for Large Scale Industrial Engineering Problems. *Knowledge-Based Systems*, 165, 169-196.

[33] S. M. Macdonald and C.F. Mason. Predation of migrant birds by gulls, Brit. Birds, 66:361–363, 1973.

[34] Kamiya, N. (1940). THE CONTROL OF PROTOPLASMIC STREAMING, *Science*, 92. 462-463.

[35] Nakagaki, T., H. Yamada, and T. Ueda. (2000). Interaction between cell shape and contraction pattern in the Physarum plasmodium. *Biophysical Chemistry*, 84, 195-204.

[36] Li, Shimin & Chen, Huiling & Wang, Mingjing & Heidari, Ali Asghar & Mirjalili, Seyedali. (2020). Slime mould algorithm: A new method for stochastic optimization. *Future Generation Computer Systems*, 10.1016/j.future.2020.03.055.