Vietnamese - German University
FACULTY OF ENGINEERING
ELECTRICAL AND COMPUTER ENGINEERING

Implementation of BAT Algorithm as Maximum Power Point Tracking Technique for Photovoltaic System Under Partial Shading Conditions

Group 13

Members	ID
Lê Thị Ngọc Diệp	16090
Lê Quốc Định	10221009
Nguyễn Hải Dương	10221012

Instructor:
Prof. Bui Minh Duong

Binh Duong, Feb - 05 - 2024

Table of Content

Table of Figures

Note: **Red is importance Figure**

I. Introduction

1. Abstract

This report presents the implementation of the Bat Algorithm (BA) as a Maximum Power Point Tracking (MPPT) technique for photovoltaic systems under 4 different partial shading conditions. The BAT algorithm, inspired by the echolocation behavior of bats, is utilized to efficiently navigate the complex landscape of voltage and current in photovoltaic systems. By mimicking the bats' ability to locate the Global Maximum Power Point (GMPP), the BA optimizes the energy harvest even in the presence of partial shading. The effectiveness of the BA in tracking the MPPT is evaluated under 4 different partial shading scenarios, providing insights into its performance and applicability. The results demonstrate the BA's capability to enhance the energy harvest by effectively adapting to changing shading conditions. This report contributes to the understanding of the BAT algorithm's potential in optimizing PV systems and highlights its relevance in renewable energy applications.

2. Overview of BAT algorithm

2.1. Definition and History

BAT algorithm (BA) was presented by Yang in 2010 as a bio-inspired algorithm. Since then, it has demonstrated remarkable efficiency and has been acclaimed for its effectiveness in various applications. The Bat Algorithm (BA) draws inspiration from the natural behaviors of bats and is foreshadowed as an innovative optimization technique. This method mirrors the strategic approach bats employ to seek the most favorable solutions, analogous to a bat's pursuit of the most nourishing sustenance. Hence, the Bat Algorithm (BA) is a powerful tool utilized in the quest for optimal problem-solving, bearing testament to the extraordinary capabilities inspired by the fascinating world of bats.

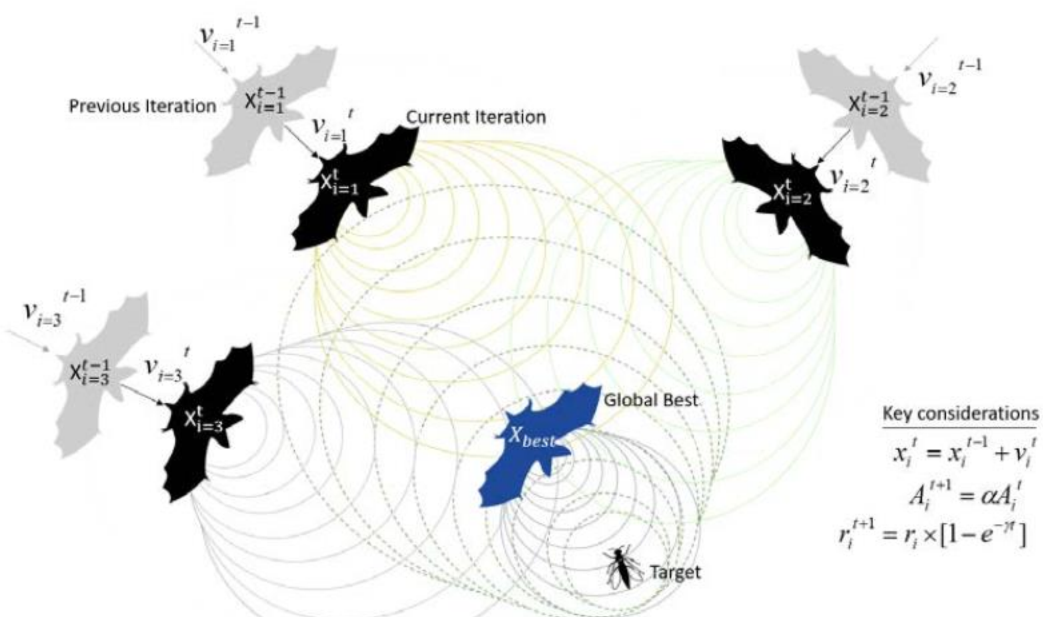


Figure 1. BAT algorithm Visualized Explanation

2.2. The Development and Widely Application of BAT algorithm.

Bat algorithms are versatile warriors, solving complex problems in situations like engineering design, choices in machine learning, and even in the complex world of finance. Effectiveness is not limited to brains but dances a delicate balance between research, breaking into and exploiting new territories, and refining existing solutions. This beautiful algorithm can navigate the search area efficiently, eventually unearthing the best solution even in the face of noisy data or incomplete information Its simple implementation and easily customizable standards make it accessible to a wide variety of users, whether veteran optimization veterans or curious newcomers. From winning complex engineering designs to optimizing finances, the BAT algorithm proves to be a powerful and approachable tool in the optimization battle.

While the bat algorithm boasts impressive strengths, its weaknesses still lurk in the shadows. First, its cybersearch can be a hungry beast, especially for large problems. This obsession limits its flexibility in real-time situations or in the face of multiple decision variables. Like a picky horse, its performance depends on carefully chosen parameters – choosing the wrong harness can result in a bumpy ride. And lastly, batter algorithms operate in the realm of heuristics, a murky world where the assurance of finding the absolute best solution is rarely a phantom. This inherent uncertainty can cause some problems to be rejected, allowing other algorithms to shine where the bat rolls. Moreover, the theoretical logic of the bat algorithm is still shrouded in fog, and its inner workings have yet to be fully elucidated. This relative invisibility can hinder its effectiveness in solving specific problems, so that its true potential in some areas is still not fully revealed Thus, when the bat algorithm is a tool powerful, we must acknowledge its limitations, lest its enthusiasm blind us to the lure of possibility.

2.3. BAT algorithm applications on finding Maximum PowerPoint for Photovoltaic Systems under Partial Shading Conditions

The Bat Algorithm takes flight beyond generic optimization, finding a particularly valuable roost in the realm of renewable energy. Under the unpredictable skies of partial shading, photovoltaic systems often lose their way, meandering from the peak power point like lost birds. But the Bat Algorithm, with its echolocation-inspired search, acts as a guiding beacon. Mimicking the bats' use of biosonar, it navigates the intricate landscape of voltage and current, expertly locating the Global Maximum Power Point (GMPP). This translates to a dramatic boost in energy harvest, squeezing every drop of potential from the sun even when shadows play hide-and-seek across the panels. Imagine you're a mountaineer trying to reach the highest peak, but clouds and fog are obscuring your view. Your compass spins uselessly, the path lost in the swirling white. Your boots crunch on loose scree, uncertainty gnawing at your resolve. Fear not, fellow mountaineer, for a secret weapon hums in your pack – the Bat Algorithm. Imagine these marvels not as tools, but as active minds each with a tiny echo station bat. They rush forward, wings beating the invisible form of the hidden mountain. Some venture into treacherous hillsides for shortcuts, while others refine their search based on prior experience and stick to promising peaks. This dynamic dance of exploration and exploitation is the tactic of these winged sailors, and they rotate regularly in the enveloped world. Now, translate this mountainous quest into sunbaked, but equally treacherous, terrain from partially shaded photovoltaic systems. The clouds, here, are dancing shadows, changing the energy harvesting height – the Global Maximum Power Point (GMPP) – in a dance of the mind. The shadows blind traditional users and cause them to stumble over the small mountains, mistaking them for real mountains. But the bat algorithm, like an experienced climber, sees through the deception. Its tiny echo bats, reading energy and electricity, fly off the invisible shape of the power source. Some venture into shady spots, relentlessly testing trails, while others stick to promising sunlit mountains, honing their search with each stroke and mimicking the sonar ballet of sharks. The resulting power increases and the evil of the shadow no longer cuts. The solar panels create a triumphant chorus, and their harvest is enhanced by an unseen choir of concerted minds. So, the next time your search for optimal energy is overshadowed by uncertainty, remember the bat algorithm – your secret weapon, your echolocation guide, always flying to the pinnacle of sunlight efficiency.

3. Advantages and Disadvantages of the BAT algorithm

3.1. Advantages of BAT algorithm

Overall, the BAT algorithm is a powerful optimization tool that can be applied to a wide range of industries and fields. Its ability to imitate the echolocation behavior of bats allows it to adapt to various optimization problems, making it a versatile algorithm. This adaptability is particularly useful in finance, where optimization problems can be complex and dynamic.

One of the key advantages of the BAT algorithm is its fast convergence rate. This means that it can quickly converge into optimal solutions, saving time and resources. In industries where time is of the essence, such as manufacturing or logistics, this can be a significant advantage. The ability to quickly find optimal solutions can lead to increased efficiency and cost savings.

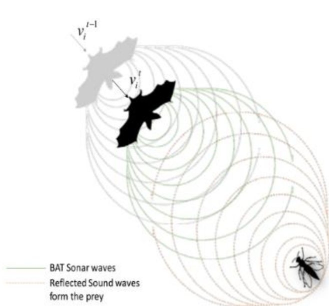
Furthermore, the BAT algorithm's nature-inspired optimization approach can be particularly effective in solving complex problems. By mimicking the behavior of bats, the algorithm can explore and exploit the search space uniquely. This can lead to innovative and creative solutions that may not be found using traditional optimization methods.

3.2. Disadvantages of BAT algorithm

However, there are drawbacks to consider when using the BAT algorithm. One of the main limitations is its heavy reliance on randomization. While randomization can help the algorithm explore the search space and avoid getting stuck in local optima, it can also lead to suboptimal results. The algorithm may not always find the absolute best solution, and there is a trade-off between exploration and exploitation that needs to be carefully considered.

Another challenge with the BAT algorithm is parameter tuning. To ensure the algorithm's effectiveness, the parameters must be carefully selected and tuned. This process can be time-consuming and challenging, as finding the right combination of parameters greatly impacts the algorithm's performance. It requires a deep understanding of the problem at hand and expertise in optimization techniques.

Figure 2. BAT Visualization



II. Technical Search

1. How the BAT algorithm works

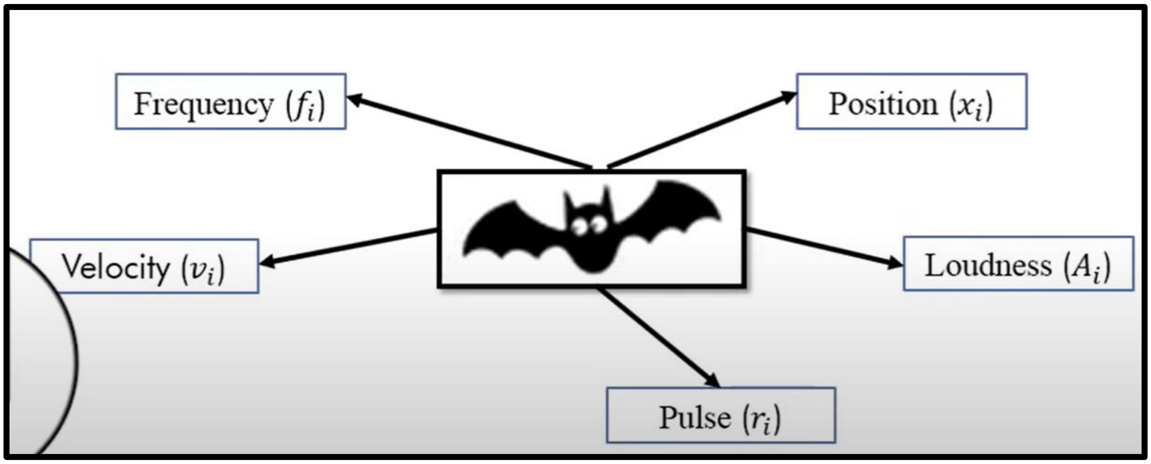


Figure 3. BAT algorithm Parameters

Key Inspiration:

The Bat Algorithm (BA) is a metaheuristic optimization algorithm that draws inspiration from the echolocation behavior of microbats. It mimics how bats use varying pulse rates and loudness to navigate, detect prey, and avoid obstacles in dark environments.

Initialization:

+ Define the problem and its parameters (search space, fitness function).
+ Create a population of virtual bats, each representing a potential solution.
+ Assign initial positions, velocities, and frequencies to each bat.

Movement and Echolocation:

+ Each bat moves within the search space using a combination of:
  - Current position and velocity.
  - Frequency-tuning technique (inspired by bat echolocation).
  - Bats emit virtual "pulses" and adjust their frequencies and loudness based on the quality of solutions found (analogous to prey proximity).

Local Search and Random Flight:

+ To balance exploration and exploitation, the algorithm incorporates:
  - Local search: Bats randomly explore promising solutions.
  - Random flight: Bats occasionally take long-distance flights to discover new areas.

Loudness and Pulse Rate Adjustment:

+ As bats approach potential solutions, they decrease loudness and increase pulse emission rate (mimicking bat behavior when closing in on prey).

Selection of the Best:

+ The best solutions (bats with the highest fitness values) are identified and selected in each iterations. These best solutions influence the movement of other bats in subsequent iterations.

Termination:

+ The algorithm continues until a stopping criterion is met (e.g., maximum iterations, desired fitness level reached).

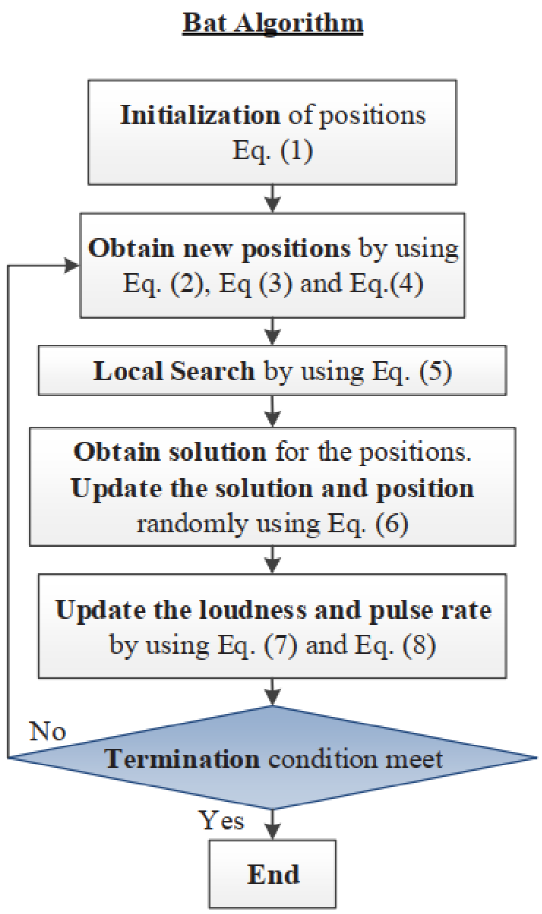


Figure 4. Flowchart BAT algorithm and Implementation of BA as MPPT

2. Code Working

2.1. Flowchart of BAT process.

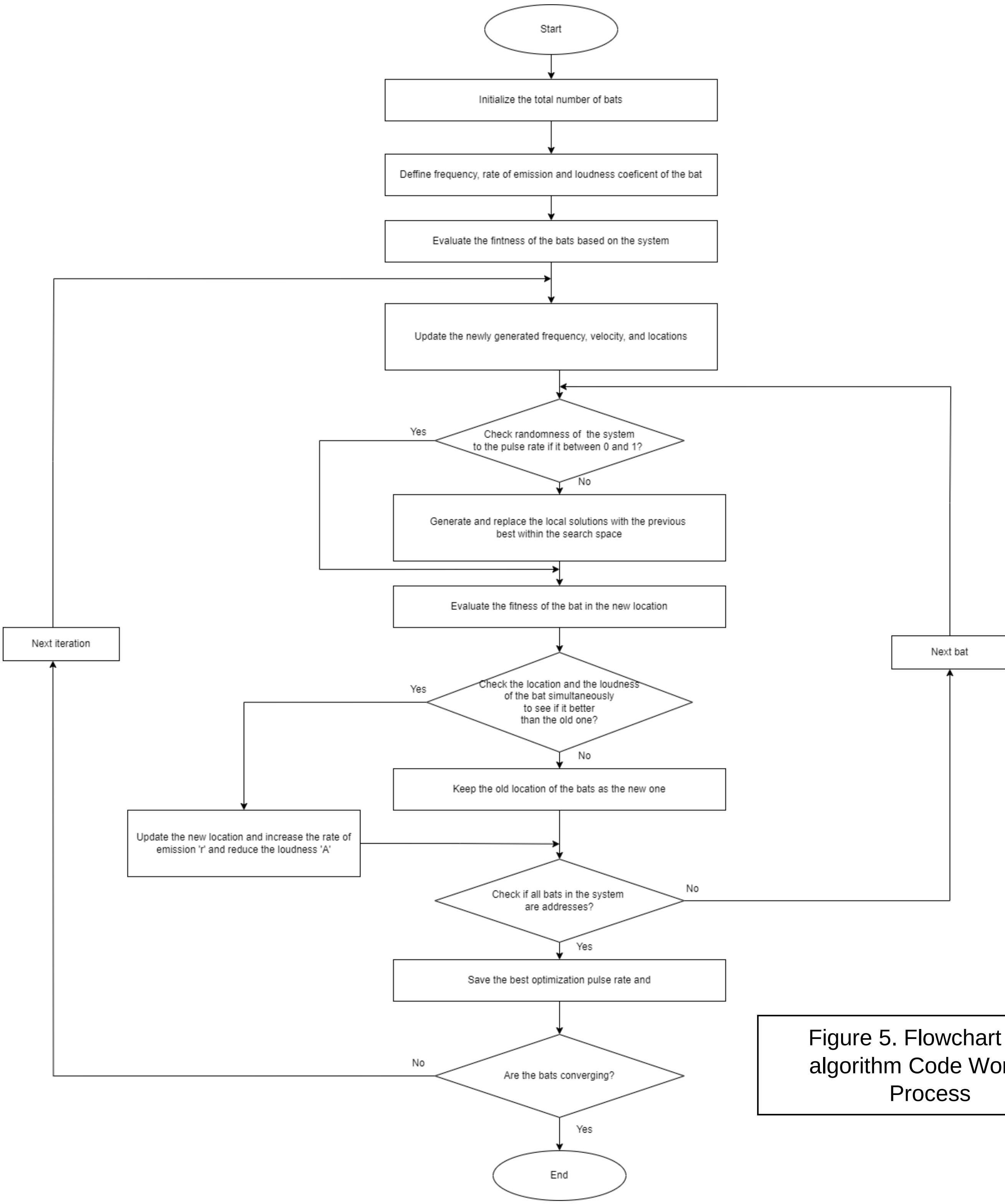


Figure 5. Flowchart BAT algorithm Code Working Process

2.2. Code BAT Function.

```
function D = BAT(Vin,Iin)

coder.extrinsic('randi')

persistent u;
persistent i;
persistent j;
persistent k;                    % Iterations
persistent DP;                   % DutyCycle
persistent DPcurrent;            % Current DutyCycle
persistent v;                    % Velocity
persistent F;                    % Frequency
persistent Fmax;                 % Maximum frequency
persistent Fmin;                 % Minimum frequency
persistent A;                    % Loudness
persistent r;                    % Pulse emission rate
persistent bestDP;
persistent DPcontainer;
persistent DPfitness;
persistent P;                    % Power
persistent Pfitness;
persistent counter;              % Counter
persistent lb;                   % Lower boundary
persistent ub;                   % Upper boundary
persistent ro;                   % Initial pulse emission rate
persistent alpha;                % Constant for loudness update
persistent gamma;                % Constant for emission rate update
persistent eps;
persistent Pprev;
persistent bool;
persistent closeness;
persistent index;

%initialization
if (isempty(counter))
   counter=0;
   bool=0;
   index=0;
   closeness=false;
   Pprev=Vin*Iin;
   k=0;                         % Iterations
   lb=0;                        % Lower boundary
   ub=1;                        % Upper boundary
   DPcurrent=0.5;               % Current DutyCycle
   DPcontainer=zeros(4,1);
   DPfitness=zeros(4,1);
   Fmax=0.5;                    % Maximum frequency
```

```
    Fmin=0.3;                          % Minimum frequency
    A=rand(4,1);                       % Loudness for each DutyCycle
    r=rand(4,1);                       % Pulse emission rate for each DutyCycle
    u=0;
    i=0;
    v=zeros(4,1);                     % Velocities
    F=zeros(4,1);                     % Frequency
    P=zeros(4,1);                     % Power
    Pfitness=zeros(4,1);
    alpha=0.5;                        % Constant for loudness update
    gamma=0.5;                        % Constant for emission rate update
    ro=0.67;                          % Initial pulse emission rate
    DP=zeros(4,1);                    % Dutycycles
    %initial dc for each particle
    DP(1)=0.2;
    DP(2)=0.4;
    DP(3)=0.6;
    DP(4)=0.8;
end


% At the first time, this if ignored (counter=0)
% Delay
if(counter>=1 && counter<300)
    D=DPcurrent;
    counter=counter+1;
    return;                    % return control to the invoking function before it reaches the end of the
function.
end
counter=0;                     % reset the counter

% calculate the fittnes function (power) of each particle
% then compare the current value of the function with the previous one
% if the current value is better than the previous, update the value
% Note: at the first time, this if ignored (u=0)
if(u>=1 && u<=4)
    P(u)=Vin*Iin;
    DPcontainer(u)=DPcurrent;
end

% Update if the solution improves, or not too loud
if(u>4)
    for i=1:4
        if (P(i)>=Pfitness(i))
            Pfitness(i)=P(i);
            DPfitness(i)=DPcontainer(i);
            if(k>=1)
                A(i)=alpha*A(i);
                r(i)=ro*(1-exp(-gamma*k));
            end
```

```
        end
    end
end

u=u+1;
% AT the first time, this is excuted because it consist the condition...
% u==0 (initial value of u)
if(u==6)
    u=1;
end

if(u==5)
    if(bool==0)
        k=k+1;
    end

    [Pmax,index]=max(Pfitness);     % finds the indices of the maximum values of P
                                    % (max power) and returns them in output vector "index"
    bestDP=DPfitness(index);        % find the location(duty) of the particle which has max Power
    if(bool==0)
        D=bestDP;
    else
        D=DP(index);                %mean
    end
    DPcurrent=D;
    counter=1;

    for i=1:4
        F(i)=Fmin+(Fmax-Fmin)*rand;                   %randomly chose the frequency
        v(i)=v(i)+(DPfitness(i)-bestDP)*F(i);         %update the velocity
        if DPfitness(i)>bestDP
            DP(i)=DPfitness(i)-abs(v(i));             %update the Dutycycle position
        else
            DP(i)=DPfitness(i)+abs(v(i));             %update the Dutycycle position
        end

        %Checking bounds/limits
        if DP(i) < lb % Check the lower limit
            DP(i) = lb;
        elseif DP(i) > ub                             % Check the upper limit
            DP(i) = ub;
        end

        %check the condition with r
        if rand>r(i)
            eps=-1+(1-(-1))*rand;
            DP(i)=bestDP+eps*mean(A);
        end
    end
```

```
if(bool==0)
   closeness=true;
   for i=1:4
      for j=1:4
         if(i~=j)
            closeness = closeness & (abs(DP(i)-DP(j))<0.1);      %0.05
         end
      end
   end
end

%if the power stables already
if(bool==1 )
   %check if the power changes
   if(abs(Vin*Iin-Pprev)/Pprev >= 0.05)                 %0.1

      DPcurrent=0.5;                                    % Current DutyCycle
      DPcontainer=zeros(4,1);
      DPfitness=zeros(4,1);
      A=rand(4,1);                                      % Loudness for each DutyCycle
      r=rand(4,1);                                      % Pulse emission rate for each DutyCycle
      v=zeros(4,1);                                     % Velocities
      F=zeros(4,1);                                     % Frequency
      P=zeros(4,1);                                     % Power
      Pfitness=zeros(4,1);
      DP=zeros(4,1);                                    % Dutycycles

      DP(1)=randi([5 250])/1000;
      DP(2)=randi([250 500])/1000;
      DP(3)=randi([500 750])/1000;
      DP(4)=randi([760 995])/1000;

      bool=0;
   end
end

if(closeness==1)
   bool=1;
   closeness=false;
end

if(k>100)

      DPcurrent=0.5;                                    % Current DutyCycle
      DPcontainer=zeros(4,1);
      DPfitness=zeros(4,1);
      A=rand(4,1);                                      % Loudness for each DutyCycle
      r=rand(4,1);                                      % Pulse emission rate for each DutyCycle
      v=zeros(4,1);                                     % Velocities
      F=zeros(4,1);                                     % Frequency
      P=zeros(4,1);                                     % Power
```

```
          Pfitness=zeros(4,1);
          DP=zeros(4,1);                                          % Dutycycles

          DP(1)=randi([5 250])/1000;
          DP(2)=randi([250 500])/1000;
          DP(3)=randi([500 750])/1000;
          DP(4)=randi([760 995])/1000;

          k=0;
    end

    Pprev = Vin*Iin;

    return;
else
    %update each particle's duty cycle, reset counter.
    if(bool==0)
        D=DP(u);
    else
        D=DP(index); %mean
    end
    DPcurrent=D;
    return;
end
end
```

2.3. Explain Code - PseudoCode.

Step 1: Initialize 4 DutyCycles

$$DP_i^0 = 0.2, 0.4, 0.6, 0.8$$

+ Setup parameters:

- F_{min} = 0.3 and F_{max} = 0.5 % max and min Frequency
- alpha = 0.5 % Constant for loudness update
- gamma = 0.5 % Constant for emission rate update
- ro = 0.67 % Initial pulse emission rate

Step 2: Create new DutyCycles

$$f_i = 0.3 + (0.5 - 0.3)\beta$$

$$v_i^{t+1} = v_i^t + (DP_i^t - DP_{best}^t) f_i$$

$$DP_i^{t+1} = \begin{cases} DP_i^t - |v^{t+1}|; & if \quad DP_i^t > D_{best} \\ DP_i^t + |v^{t+1}|; & else \end{cases}$$

Step 3: Checking boundary.
+ Lower_bound<=DP<=Upper_bound
+ with ub = 1 and lb =0

Step 4: Apply local search.

$$if(rand(0,1) > r_i^t$$
$$DP_i^{t+1} = DP_i^t + \varepsilon * A_{mean}^t$$

Step 5: Compare with the previous good value.

$$if(P(x_i^{t+1}) < P(x_i^t))$$
$$DP_i^{t+1} = DP_i^t$$
$$P_i^{t+1} = P_i^t$$

Step 6: Update stronger loudness A(i) and pulse rate r(i) after condition 5 is met.

$$A_i^{t+1} = 0.5 A_i^t$$
$$r_i^{t+1} = r_i^0(1 - e^{-0.5t})$$

Step 7: Checking result of each DutyCycle

+ closeness = closeness & (abs(DP(i)-DP(j))<0.1);
- ➔ If: closeness = true;
  - ➢ If: (abs(Vin*Iin-Pprev)/Pprev >= 0.05);
    - ✓ DutyCycle have been found is good.
  - ➢ Else
    - ✓ DutyCycle is not good.
    - ✓ Break and return the loop with new dutycycles.
- ➔ Else: Break and return the loop with new dutycycles.

Step 8: Loop Running

- All the functions in running in the loop forever correspond to the time interval setup in MATLAB Simulink.
- And each Duty cycle is shown whenever the loop function come to them, and the Power is coming along with.
- The Loop with moving to update duty cycle function whenever **"u == 5"**.

<u>Note:</u> Vin and Iin are consequently updated to correspond with the DutyCycles and Powers during the loop function.

3. Circuit Design

3.1. PV arrays Setup

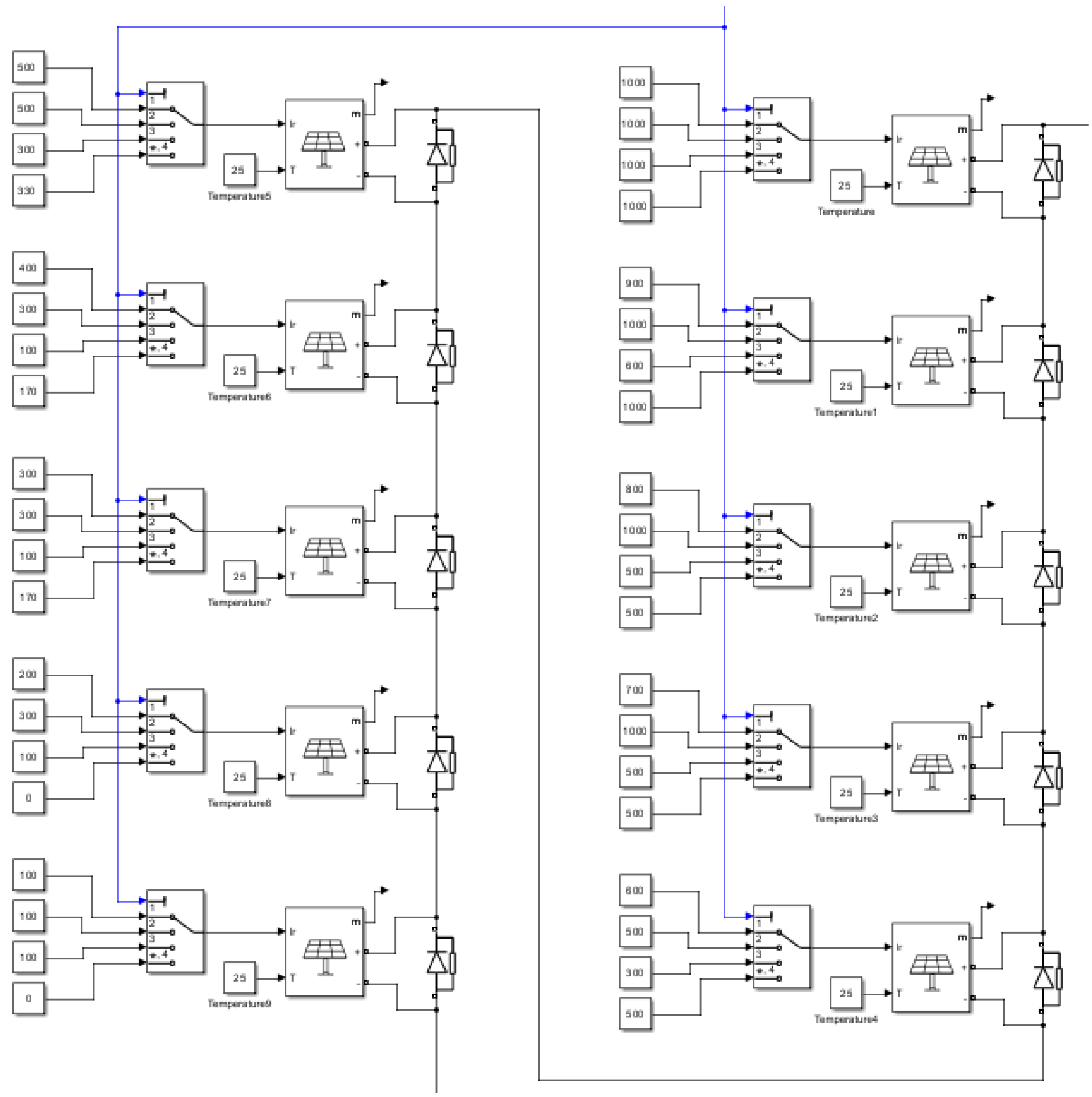


Figure 6. 10 Series PV array Setup

This is 10 PV-array connected in series with 4 different Partial Shading Conditions (Condition A, Condition B, Condition C, Condition D) which would be controlled under the rotating switch:

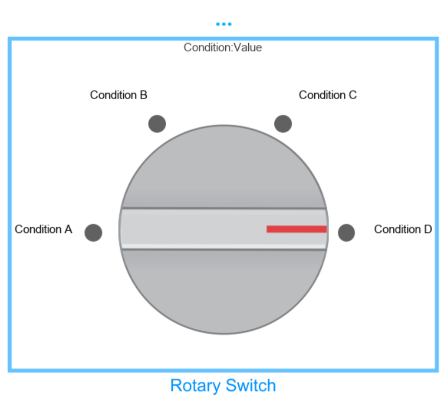


Figure 7. Rotary Switch Controlled Conditions

Partial Shading Conditions

Condition A: Each module gradually decreased from 1000W/m^2 to 100W/m^2.

Condition B: 4 modules at 1000W/m^2, 2 modules at 500W/m^2, 3 modules at 300W/m^2 and a module at 100W/m^2.

Condition C: 1 module at 1000W/m^2, 1 module at 600W/m^2, 2 modules at 500W/m^2, 2 modules at 300W/m^2 and 4 modules at 100W/m^2.

Condition D: 2 modules at 1000W/m^2, 3 modules at 500W/m^2, 1 module at 330W/m^2, 2 modules at 170W/m^2 and 2 modules are fully shaded.

3.2. Ramp Function Circuit and PV Curve.

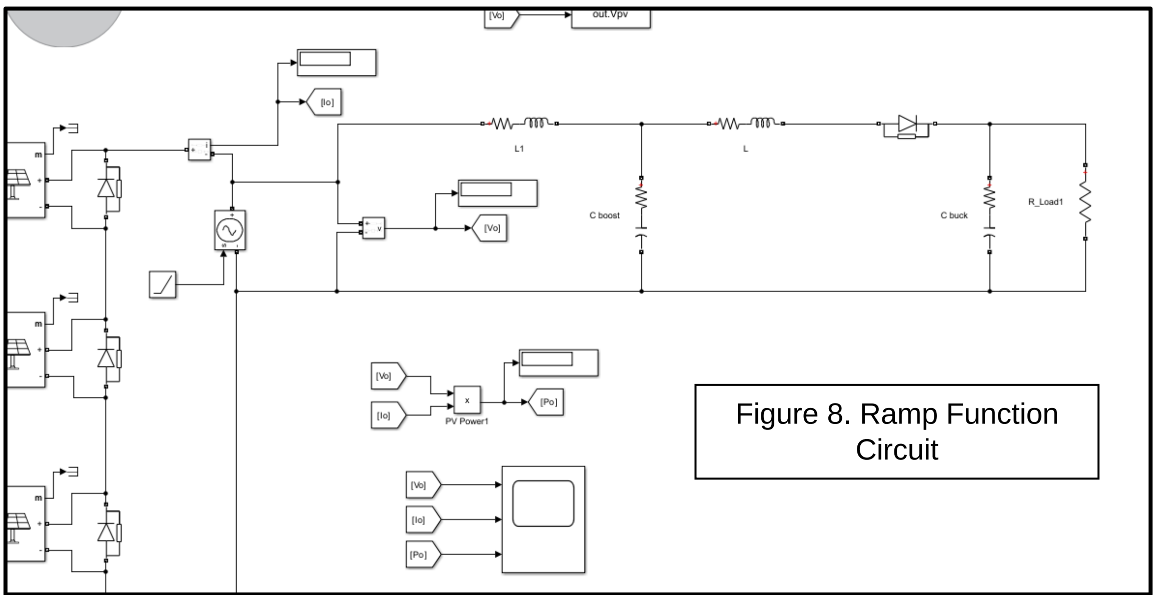
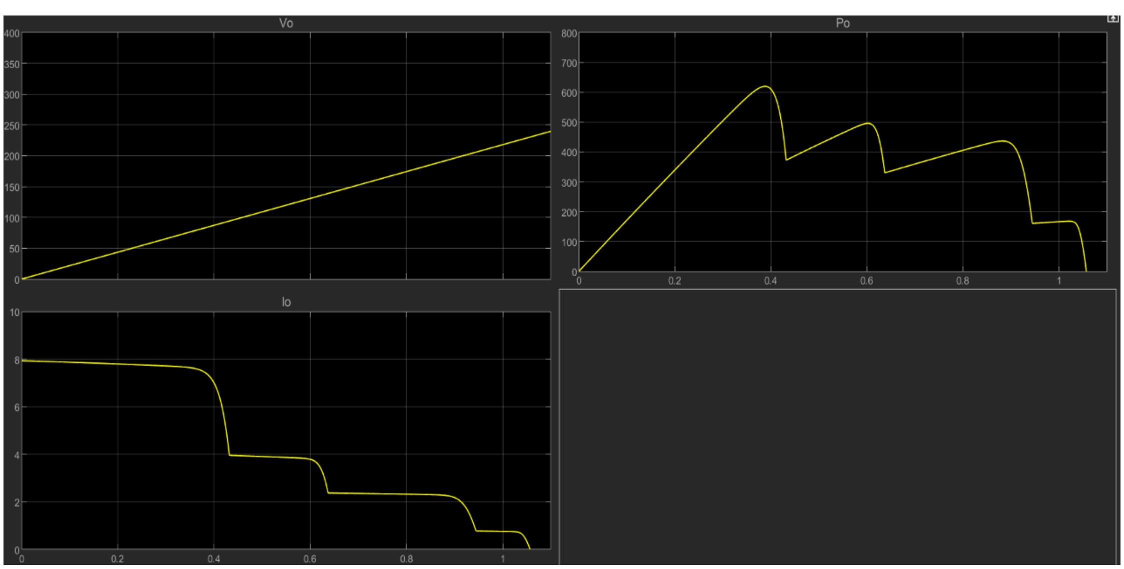


Figure 8. Ramp Function Circuit

The **Ramp function,** which is applied to the circuit above, will increase the input Voltage linearly in operation range. The input Voltage increase resulted in the decrease in Current and created a **Curve of Power**. (**Note:** Picture below is just a random example)



3.3. BAT algorithm applied code Circuit and tracking

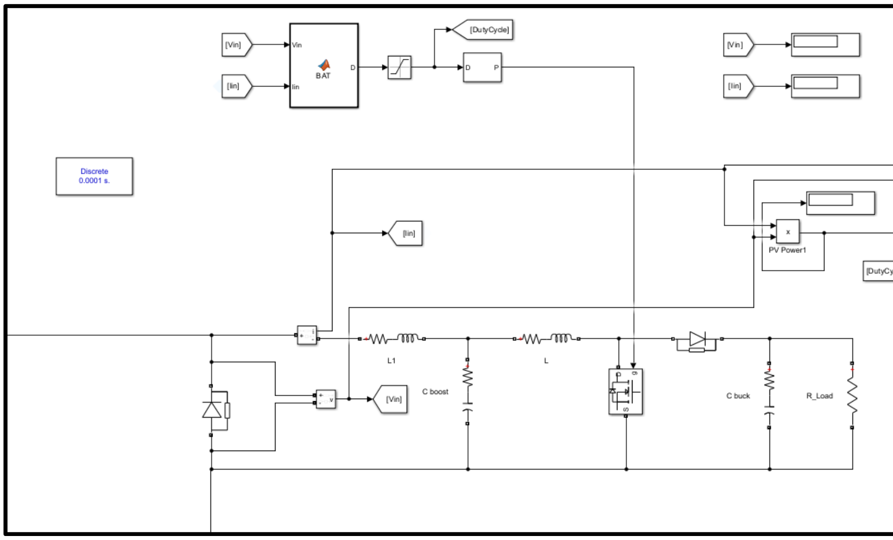
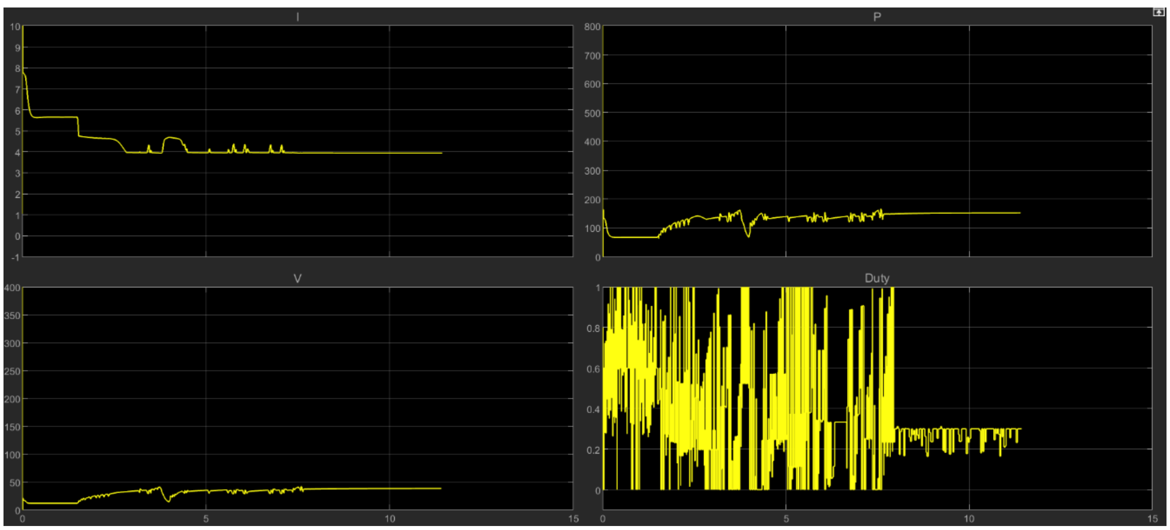


Figure 9. BAT algorithm applied code Circuit

The BAT algorithm embedded in BAT function Block would consequently update the value of Vin and Iin to calculate the Power. Then, new duty cycles will be generated with the target of finding out the **Maximum Power**. (**Note:** Picture below is just a random example)



PV module and Circuit Parameters.

PV module: Mitsubishi PV-AE125MF5N

TABLE I. PARAMETER OF PV-AE125MF5N PV MODULE AT STC TEMPERATURE = 25°C, INSOLATION = 1000 W/M²

Maximum Power Rating (Pmax)	125 W
Open Circuit Voltage (Voc)	21.8 V
Short Circuit Current (Isc)	7.9 A
Maximum Power Voltage (Vmp)	17.3 V
Maximum Power Current (Imp)	7.23 A

Circuit Paramaters

L1: 1Ω, 100mH
L: 1Ω, 100mH
C_boost: 1Ω, 100mF
C_buck: 1Ω, 150uF

R_load: 30Ω

Figure 10. PV module and Circuit Elements - Parameters

4. MPPT and DutyCycle Result

4.1. Result

We employ a highly sophisticated and advanced algorithmic approach to meticulously generate the maximum possible value and potential impact of a PowerPoint presentation (MPPT). With our cutting-edge technology, we meticulously analyze and optimize every aspect of the presentation to ensure its effectiveness and success. Additionally, we utilize the highly effective ramp function, which allows us to carefully and meticulously sketch the curve of the presentation. This process ensures a comprehensive examination and evaluation of the two aforementioned MPPT values, taking into consideration various factors such as visual appeal, content organization, and audience engagement. Through this meticulous validation process, we ensure the congruence and alignment of the two MPPT values, thereby enhancing the overall reliability, accuracy, and success rate of our algorithmic system.

Furthermore, it is important to note that our sophisticated algorithmic approach takes into account four distinct results that are compatible with four specific conditions. By carefully analyzing these conditions, we can optimize the presentation to meet the unique requirements and preferences of each situation. This customized approach ensures that the generated PowerPoint presentation is tailored to the specific needs and objectives of our clients, maximizing its potential to communicate the desired message and achieve the desired outcomes.

In conclusion, our comprehensive approach to PowerPoint presentation generation combines sophisticated algorithms, meticulous analysis, and the utilization of advanced functions to create impactful and successful presentations. By meticulously examining and evaluating the maximum ppt values, we guarantee the congruence of these values, which enhances the reliability and accuracy of our algorithmic system. With our tailored approach and attention to detail, we ensure that the generated presentations are optimized to meet the unique requirements of different situations, resulting in effective communication and desired outcomes.

Table 1. Voltage, current and impedance equations of DC-DC converters.

Type of Controller	Voltage Equations	Current Equations	Impedance Equations
Buck	$V_{in} = \frac{1}{D} \cdot V_{out}$	$I_{in} = D \cdot I_{out}$	$Z_{in} = \frac{1}{D^2} \cdot Z_{out}$
Boost	$V_{in} = (1-D) \cdot V_{out}$	$I_{in} = \frac{1}{(1-D)} \cdot I_{out}$	$Z_{in} = (1-D)^2 \cdot Z_{out}$
Buck-boost	$V_{in} = -\frac{(1-D)}{D} \cdot V_{out}$	$I_{in} = -\frac{D}{(1-D)} \cdot I_{out}$	
Ćuk	$V_{in} = -\frac{(1-D)}{D} \cdot V_{out}$	$I_{in} = -\frac{D}{(1-D)} \cdot I_{out}$	$Z_{in} = \frac{(1-D)^2}{D^2} \cdot Z_{out}$
SEPIC	$V_{in} = \frac{(1-D)}{D} \cdot V_{out}$	$I_{in} = \frac{D}{(1-D)} \cdot I_{out}$	

SEPIC: single-ended primary-inductor converter.

Figure 11. Relationship table between Input and Output Setup-Parameters

Condition A

Each module gradually decreased from 1000W/m^2 to 100W/m^2

BAT tracking MMPT and P-V Curve

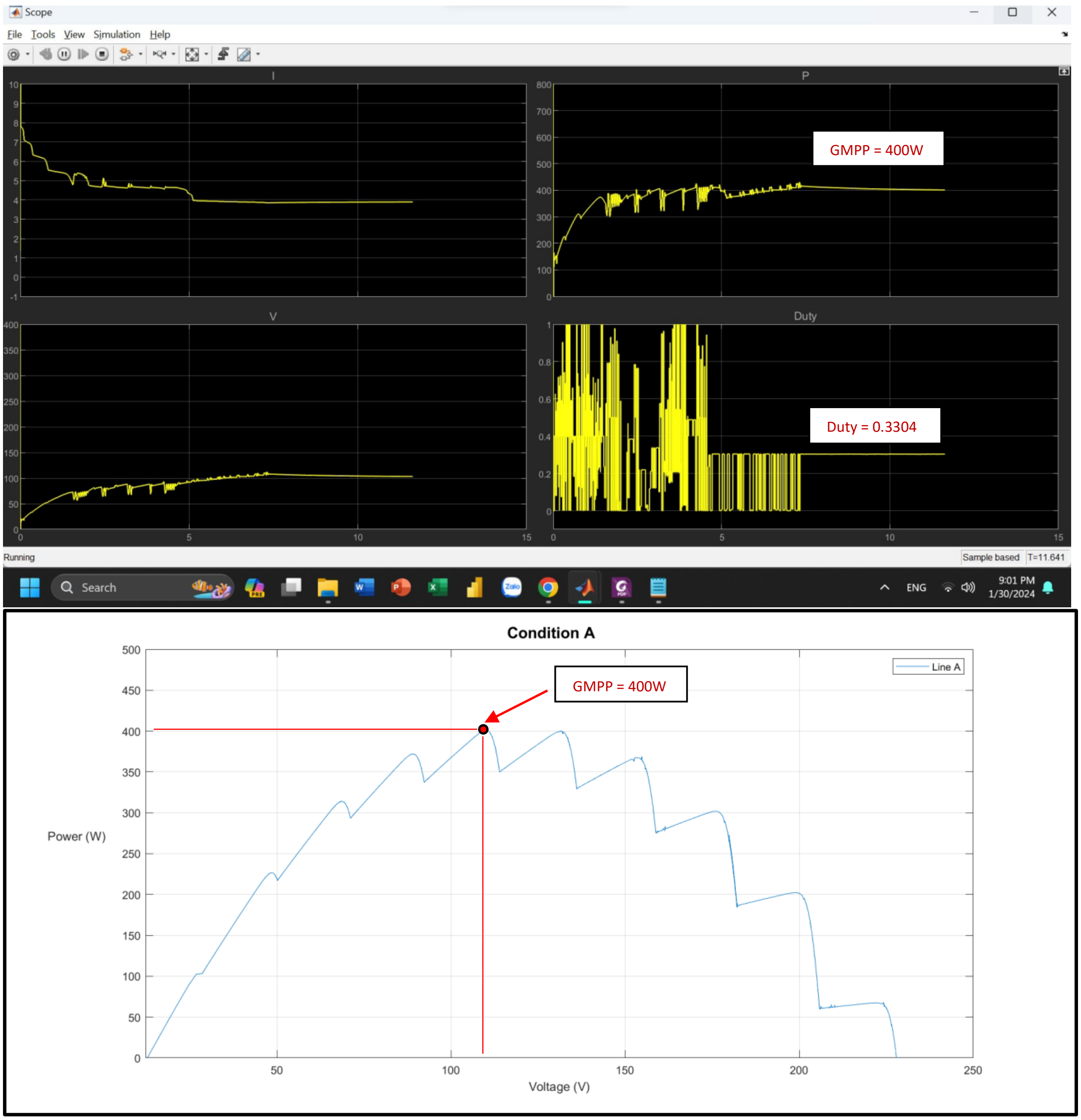


Figure 12. Partial Shading Condition A

Condition B

4 modules at 1000W/m^2, 2 modules at 500W/m^2, 3 modules at 300W/m^2 and a module at 100W/m^2

BAT tracking MMPT and P-V Curve

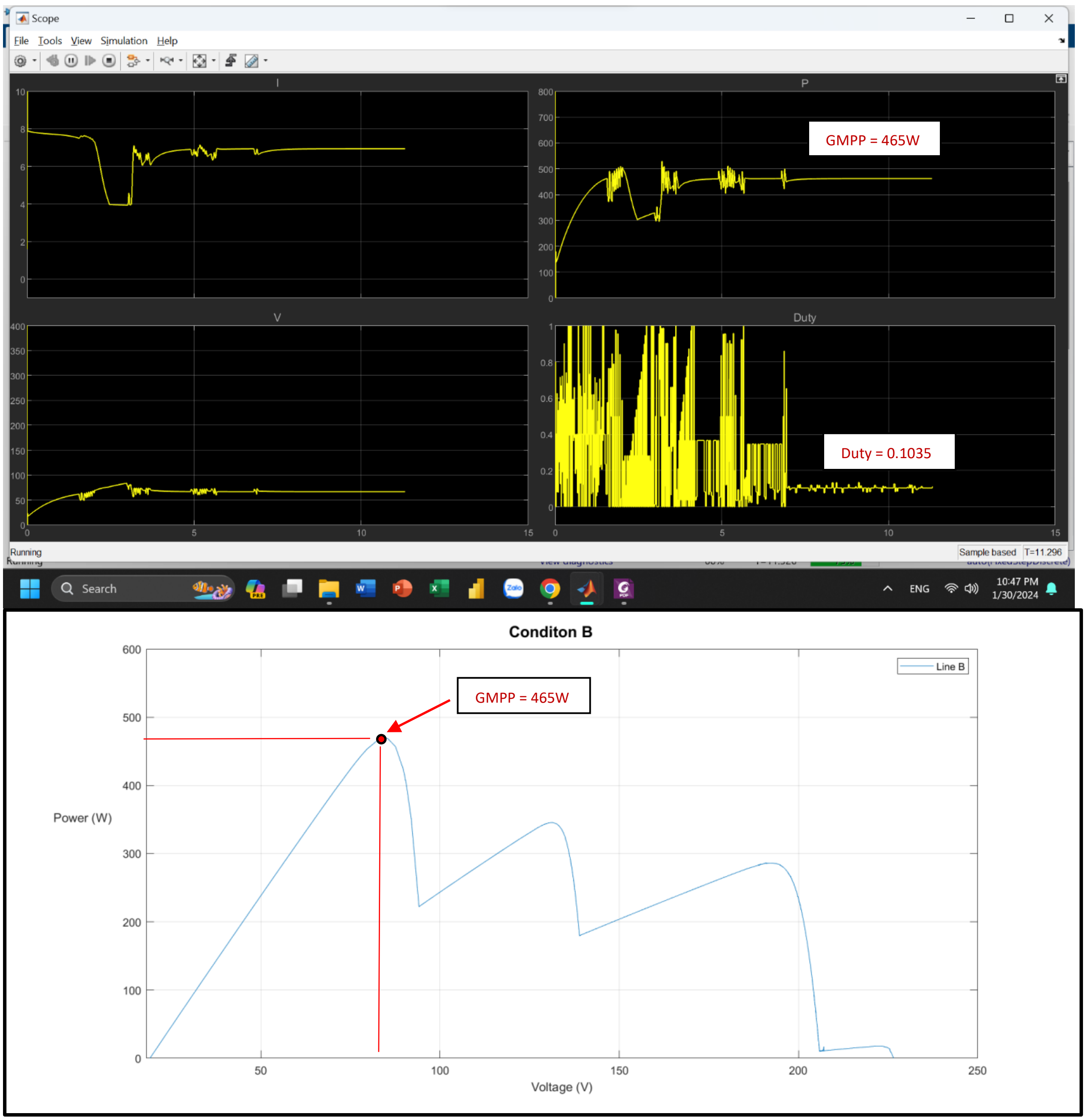
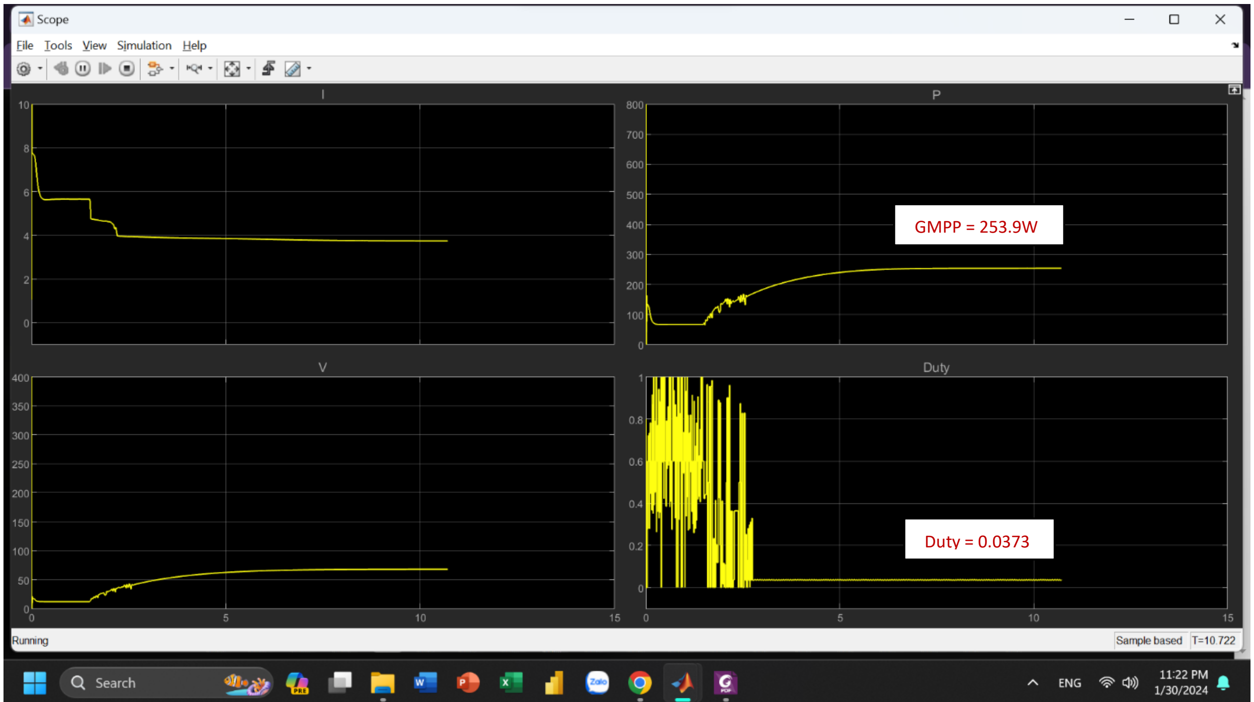


Figure 13. Partial Shading Condition B

Condition C

1 module at 1000W/m^2, 1 module at 600W/m^2, 2 modules at 500W/m^2, 2 modules at 300W/m^2 and 4 modules at 100W/m^2

BAT tracking MMPT and P-V Curve



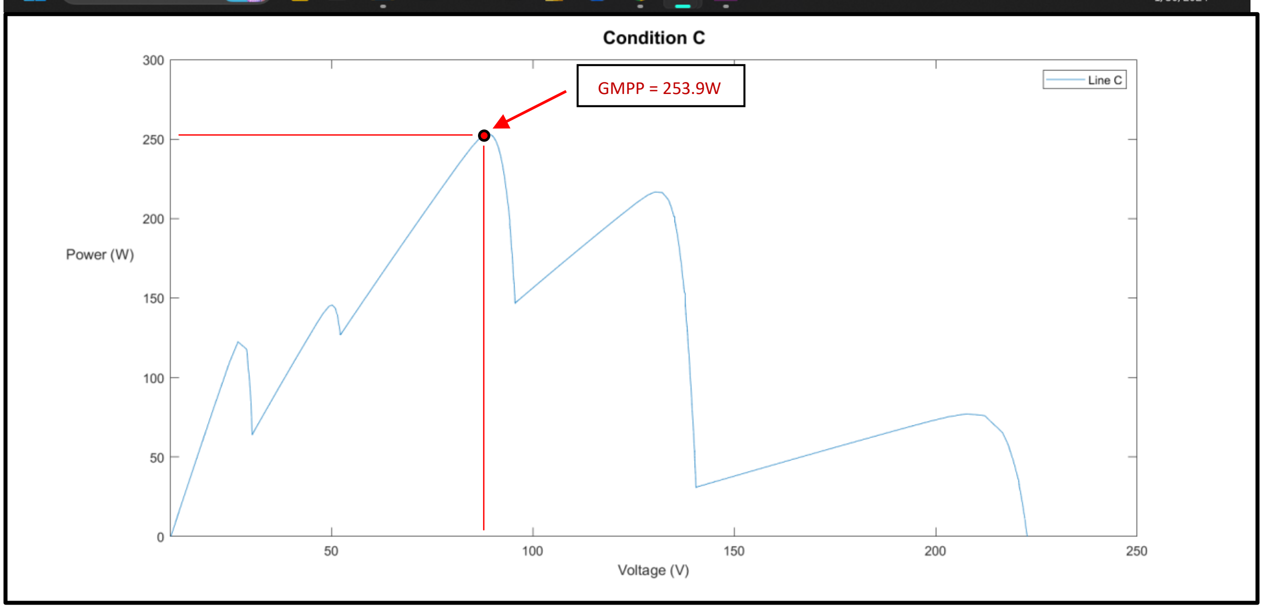


Figure 14. Partial Shading Condition C

Condition D

2 modules at 1000W/m^2, 3 modules at 500W/m^2, 1 module at 330W/m^2, 2 modules at 170W/m^2 and 2 modules are fully shaded.

BAT tracking MMPT and P-V Curve

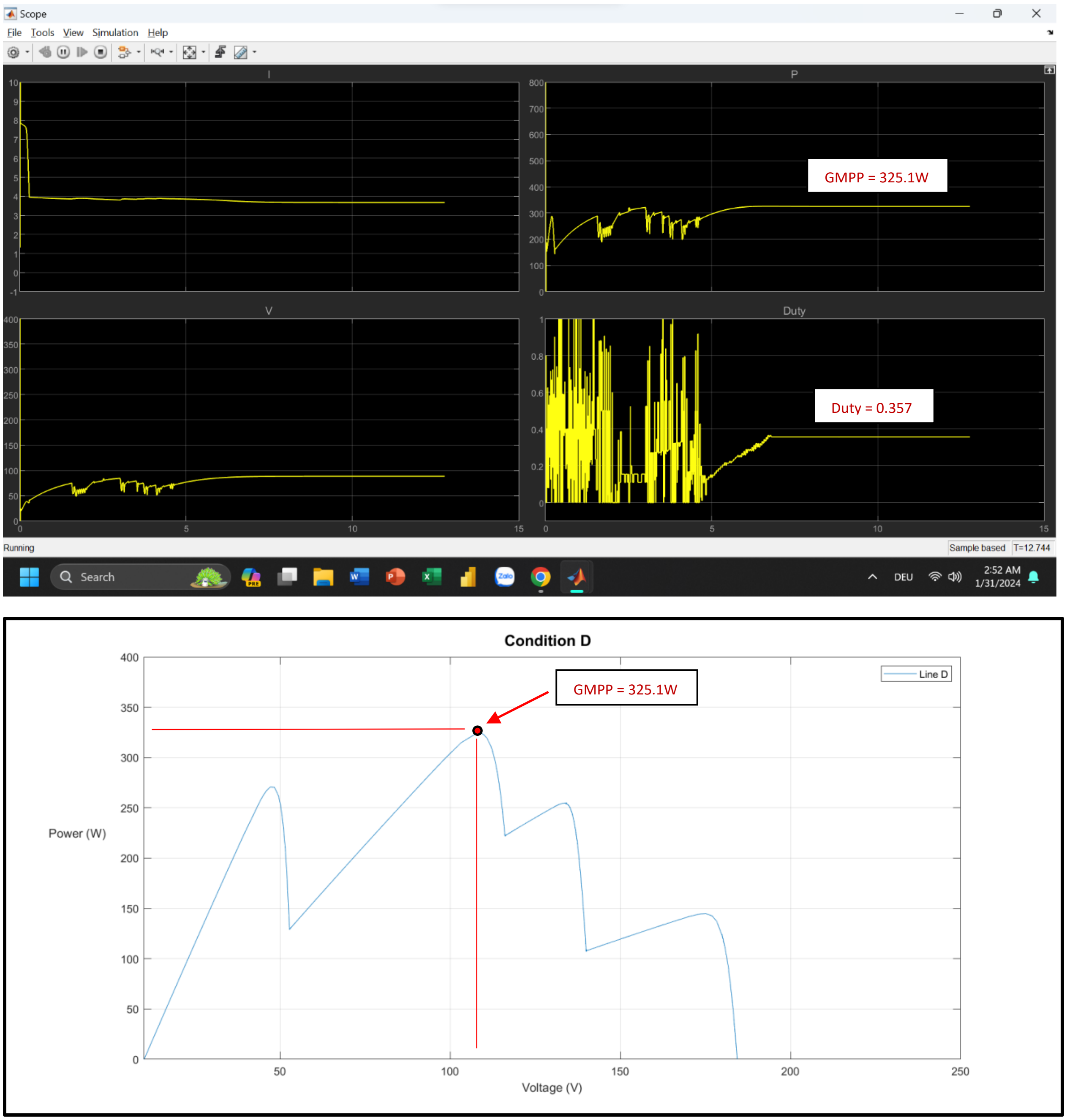


Figure 15. Partial Shading Condition D

4.2. Result Discussion

+ GMPP tracking capabilities: High

+ Efficiency or Accuracy: Very High (It usually tracks out the GMPP as compared to the **P-V Curve**)

+ Speed: High (usually around smaller than 8s)

+ Reliability: High

+ Response to load variation: Fast

+ Steady state oscillation: No oscillation when finding out the best duty cycle.

Discussion

The result of implementing the Bat Algorithm (BA) as a Maximum Power Point Tracking (MPPT) technique under four different partial shading conditions provides valuable insights into its performance and effectiveness. In this section, we discuss several key observations and findings from the experimental results.

Firstly, it was observed that the BA exhibited robust performance across all four partial shading scenarios. The algorithm demonstrated its ability to adapt and dynamically track the GMPP even in the presence of varying shading conditions. This is a significant advantage, as partial shading can significantly impact the energy harvest of photovoltaic systems. The BA's adaptive nature allowed it to effectively navigate through these challenges and optimize the energy extraction from the solar panels.

Furthermore, the BA showcased its capability to quickly converge to optimal solutions. The fast convergence rate of the algorithm was particularly evident in scenarios where the shading conditions changed rapidly. This attribute is crucial in real-time applications, where swift adjustment to fluctuations in shading is necessary to maintain optimal power output. The BA's ability to rapidly track the GMPP in such dynamic scenarios contributes to increased energy efficiency and improved system performance.

Another noteworthy observation is the BA's resilience to local optima. Partial shading conditions often introduce local peaks and valleys in the power-voltage curve, making it challenging for traditional MPPT techniques to accurately track the GMPP. However, the BA's exploration and exploitation capabilities enabled it to overcome this challenge. By leveraging its stochastic search strategy and adaptability, the BA was able to navigate through the search space and identify the true GMPP, even in the presence of local optima. This ability to overcome local optima is a significant advantage of the BA and sets it apart from conventional MPPT techniques.

Moreover, the BA demonstrated its versatility in handling different levels of partial shading. The experimental results showed that the algorithm was effective in scenarios ranging from mild to severe partial shading. This versatility is crucial in practical applications, as shading conditions can vary widely depending on factors such as time of day, weather conditions, and surrounding objects. The BA's ability to adapt to different levels of shading ensures robust MPPT performance across various environmental conditions.

However, it is important to note some limitations and areas for further investigation. One limitation is the reliance of the BA on parameter tuning. The performance of the algorithm is highly dependent on the appropriate selection and optimization of its parameters. In this study, we employed empirically determined parameter values, but further research could explore more sophisticated techniques for parameter tuning to enhance the BA's performance.

Additionally, while the BA showcased promising results in tracking the GMPP under partial shading conditions, further comparative studies with other MPPT techniques would be beneficial. This would provide a comprehensive analysis of the BA's performance in relation to existing methods and enable a more informed assessment of its strengths and weaknesses.

In conclusion, the experimental results demonstrate the effectiveness of the Bat Algorithm as a Maximum Power Point Tracking technique under various partial shading conditions. The BA's adaptive nature, fast convergence rate, resilience to local optima, and versatility make it a promising solution for optimizing the energy harvest of photovoltaic systems. Further research and development in parameter tuning and comparative analysis will contribute to unlocking the full potential of the BA in MPPT applications.

Reference Table

Table 4. Output results of BA algorithm under three PS conditions.

Evaluated Parameter	Incremental Conductance [6]	DIRECT [3]	P&O [7]	PSO [11]	DE [12]	DEPSO [14]	Proposed Method (BA)
GMPP tracking capabilities	Low	Moderate	High	High	High	High	High
Efficiency	Low (Under PSC)	High	High	High	High	High	Very High
Simplicity	Simple	Moderate	Moderate	Moderate	Moderate	Moderate	Moderate
Speed	Very High	Moderate	Moderate	Moderate	High	Moderate	High
Reliability	Low	Moderate	High	High	High	Moderate	High
Location dependency	Yes	No	Yes	Yes	No	No	No
Steady-state oscillation	Yes	No	No	No	No	No	No
Response to load variation	Moderate	Slow	Slow	Slow	Fast	Fast	Fast
Tuning Dependency	Low	High	High	High	Moderate	Moderate	Moderate

Figure 16. Table Compare Algorithms.

III. Conclusion

To summarize, the BAT algorithm has truly brought to light remarkably effective and efficient solutions that were previously concealed from view, thereby uncovering a wealth of untapped potential on a grand scale. Through its unique and innovative methodology, this algorithm demonstrates that the most optimal and advantageous answers often lie in the tranquility of nature, where profound insights and innovative solutions can be found, contrary to what many believe. The BAT algorithm has truly revolutionized the way we approach problem-solving by shining a light on solutions that were once hidden in the shadows of obscurity. By leveraging the power of advanced computational techniques and drawing inspiration from the natural world, this algorithm has unlocked a treasure trove of possibilities that were previously overlooked.

By exploring the synergy between nature and technology, the BAT algorithm has managed to uncover connections and patterns that were previously unseen. It has shown that nature is not just a source of tranquility, but also a vast repository of knowledge and inspiration. This algorithm's ability to identify the most optimal and advantageous answers has proven to be invaluable across various fields, such as medicine, engineering, and finance.

In addition, the BAT algorithm has shed light on the untapped potential of existing data sets. By analyzing vast amounts of information and identifying underlying patterns, it has revealed new avenues for innovation and growth. This approach challenges the conventional wisdom that groundbreaking solutions require complex and convoluted processes. Instead, it highlights the inherent simplicity and beauty of nature, where elegant and efficient solutions often reside.

Moreover, the BAT algorithm has sparked a paradigm shift in problem-solving methodologies. It has encouraged researchers and practitioners to reassess their preconceived notions and embrace unconventional approaches. Through its unique and forward-thinking methodology, this algorithm has fostered a culture of curiosity and exploration. By encouraging individuals to step outside their comfort zones and venture into the depths of nature, it has paved the way for breakthrough innovations and transformative ideas.

Furthermore, the BAT algorithm's impact extends beyond the realm of problem-solving. It has inspired a renewed appreciation for the interconnectedness of the natural world and the human experience. By demonstrating the power and wisdom of nature, it has encouraged individuals to cultivate a deeper sense of harmony and respect for the environment. As more people recognize the value of natural solutions, we can expect a shift towards more sustainable and eco-friendly practices.

In conclusion, the BAT algorithm has truly revolutionized the way we approach problem-solving. Through its unique and innovative methodology, it has uncovered a wealth of untapped potential and revealed the inherent beauty and wisdom of nature. By challenging conventional wisdom and encouraging unconventional approaches, it has opened new doors for innovation and growth. The impact of this algorithm extends beyond problem-solving, inspiring a deeper appreciation for the interconnectedness of the natural world and paving the way for a more sustainable future.

IV. References

[1]: *Yang, X.-S., & Gandomi, A. H. (2012, November 28). Bat Algorithm: A novel approach for global engineering optimization. arXiv.org. https://arxiv.org/abs/1211.6663*

[2]: *Wafa Zubair Al-Dyani School of Computing, Al-Dyani, W. Z., Computing, S. of, Farzana Kabir Ahmad School of Computing, Ahmad, F. K., Siti Sakira Kamaruddin School of Computing, Kamaruddin, S. S., & Metrics, O. M. A. (1970, January 1). Improvements of bat algorithm for optimal feature selection:: A systematic literature review: Intelligent data analysis: Vol 26, no 1. Intelligent Data Analysis. https://dl.acm.org/doi/abs/10.3233/IDA-205455*

[3]: *Seyedmahmoudian, M., Kok Soon, T., Jamei, E., Thirunavukkarasu, G. S., Horan, B., Mekhilef, S., & Stojcevski, A. (2018, April 26). Maximum power point tracking for photovoltaic systems under partial shading conditions using bat algorithm. MDPI. https://www.mdpi.com/2071-1050/10/5/1347*

[4]: *Bat algorithm in discrete optimization: A review of recent ... (n.d.). https://www.semanticscholar.org/paper/Bat-algorithm-in-discrete-optimization-:-A-review-Kongkaew/c016b12d7f6930b0234aa293dbde9f7d67024e92*

[5]: *Seyedmahmoudian, M., Tey, K. S., Jamei, E., Thirunavukkarasu, G., Horan, B., Mekhilef, S., & Stojcevski, A. (1970, January 1). Maximum power point tracking for photovoltaic systems under partial shading conditions using bat algorithm. UM Research Repository. https://eprints.um.edu.my/20735/*

[6]: *Asim, M., Agrawal, P., Tariq, M., & Alamri, B. (2022, January 1). MPPT-based on bat algorithm for photovoltaic systems working under partial shading conditions. Journal of Intelligent & Fuzzy Systems. https://content.iospress.com/articles/journal-of-intelligent-and-fuzzy-systems/ifs189754*

Duty Roster

Tasks	Name	Responsible	Start-date	End-date	Status
Topic Discussion	Diep Dinh Duong	All	Nov 20-2023	Nov 25-2023	Done
Project Outline	Diep Dinh Duong	Diep	Nov 20-2023	Nov 25-2023	Done
Topic Research	Diep Dinh Duong	All	Nov 20-2023	Nov 30-2023	Done
Flowchart Drawing	Diep Dinh Duong	Dinh	Nov 25-2023	Nov 30-2023	Done
Coding Application	Diep Dinh Duong	Dinh	Dec 3 - 2023	Jan 20 - 2024	Done
Coding Optimization	Diep Dinh Duong	Dinh	23 Jan - 2024	Jan 29 - 2024	Done
Circuit Drawing	Diep Dinh Duong	Dinh Duong	Dec 15 - 2023	Jan 29 - 2024	Done
Combination of coding and circuit Research	Diep Dinh Duong	Dinh Duong	25 Jan - 2023	Jan 29 - 2024	Done
Finale Application	Diep Dinh Duong	Dinh Duong	Nov 20 - 2023	Jan 29 - 2024	Done
Project Presentation	Diep Dinh Duong	Diep	Nov 20-2023	Jan 29 - 2024	Done
Project Report	Diep Dinh Duong	All	Nov 20-2023	Jan 31 - 2024	Done
Presentation Rehearsal	Diep Dinh Duong	All	Jan 23 -2024	Jan 27-2024	Done
Report Final Review	Diep Dinh Duong	All	Feb 02 -2024	Feb 02 -2024	Done