 https://doi.org/10.24191/jeesr.v29i1.023

Performance Index-Based Evaluation of Draco Lizard Optimizer for Future MPPT Applications Under Partial Shading Conditions

Norazlan Hashim*, Mohd Huzaifah Abdullah, Khairul Safuan Muhammad, Ahmad Asri Abd Samat, and Azhan Ab. Rahman

***Abstract*—New metaheuristic optimization algorithms are introduced each year to address increasingly complex multimodal optimization problems. Consequently, selecting and evaluating a suitable optimization method for photovoltaic maximum power point tracking (MPPT) remains challenging, particularly because no standardized evaluation framework is available for this application. This study presents a performance index-based comparative evaluation of the Draco Lizard Optimizer (DLO) for global maximum power point (GMPP) identification under partial shading conditions (PSCs) as a preliminary assessment for future MPPT implementation. DLO is benchmarked against three established population-based metaheuristic algorithms: particle swarm optimization (PSO), whale optimization algorithm (WOA), and grey wolf optimizer (GWO). The proposed performance index (PI) integrates four weighted criteria relevant to assessing the suitability of metaheuristic algorithms for future MPPT applications: success rate (SR), convergence speed (CS), CPU time, and search agent standard deviation (STD), with respective weights of 0.55, 0.20, 0.15, and 0.10. Each algorithm was evaluated over 1,000 independent trials under three static PSC patterns representing left-, center-, and right-located GMPPs. DLO achieved the highest PI of 0.9350, followed by PSO, WOA, and GWO with PI values of 0.9137, 0.8718, and 0.8650, respectively. Although DLO did not attain the best value for every individual metric, it provided the strongest overall balance among GMPP identification reliability, convergence speed, computational efficiency, and search stability. These findings support the further investigation of DLO through converter-level MPPT simulation and experimental validation under dynamic partial shading conditions.**

***Index Terms*—Draco Lizard Optimizer, Global Maximum Power Point, Metaheuristic Algorithms, Partial Shading Conditions, Performance Index, Photovoltaic Systems.**

This manuscript is submitted on July 27, 2026, revised on August 29, 2026, accepted on September 1, 2026 and published on October 31, 2026.

Norazlan Hashim, Mohd Huzaifah Abdullah, and Khairul Safuan Muhammad are with the Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, Selangor, Malaysia.

Ahmad Asri Abd Samat is with the Electrical Engineering Studies, Universiti Teknologi MARA Cawangan Pulau Pinang, Kampus Permatang Pauh, Malaysia.

Azhan Ab. Rahman is with the Fakulti Teknologi dan Kejuruteraan Elektrik, Universiti Teknikal Malaysia Melaka (UTeM), Melaka, Malaysia.

*Corresponding author
Email address: azlan4477@uitm.edu.my

1985-5389/© 2026 The Authors. Published by UiTM Press. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/ses/by-nc-nd/4.0/).

I. INTRODUCTION

Photovoltaic (PV) power generation is strongly influenced by solar irradiance and temperature, both of which vary throughout the day. To maximize energy extraction, maximum power point tracking (MPPT) algorithms continuously adjust the operating point of the PV array under changing environmental conditions [1]. Under partial shading conditions (PSCs), uneven irradiance caused by clouds, buildings, trees, or other obstructions may activate the bypass diodes installed across the PV modules, allowing current to flow around the affected modules and preventing hotspot formation. This produces multiple local maximum power points (LMPPs) and a single global maximum power point (GMPP) on the power–voltage characteristic curve. Conventional MPPT methods may become trapped at an LMPP and fail to extract the available power at the GMPP, resulting in significant power loss. Population-based metaheuristic algorithms are therefore attractive because of their strong global search capability and ability to distinguish the GMPP from multiple LMPPs.

Among the most widely used population-based metaheuristic algorithms [2-5] are particle swarm optimization (PSO) [6], whale optimization algorithm (WOA) [7], and grey wolf optimizer (GWO) [8]. Their effectiveness has been demonstrated in numerous real-world optimization problems, including MPPT under PSCs. Nevertheless, new optimization algorithms continue to be developed to address increasingly complex and application-specific problems. This is consistent with the No Free Lunch theorem [9, 10], which states that no single optimization algorithm performs best across all problem domains. Therefore, selecting a suitable optimization method for MPPT remains challenging, particularly because no standardized procedure is available for evaluating newly developed algorithms for this application. To address this issue, this study proposes a performance index (PI)-based procedure for evaluating the Draco Lizard Optimizer (DLO) for potential maximum power point tracking (MPPT) applications under partial shading conditions (PSCs). As a preliminary assessment rather than a complete converter-level MPPT implementation, this work represents an offline numerical benchmarking study. The PSC scenarios were represented using pre-generated static PV lookup tables containing 10,000 PV voltage (V_{PV})–PV current (I_{PV}) data pairs in the MATLAB environment.

DLO is a newly introduced algorithm inspired by the gliding, navigation, camouflage, and adaptive maneuvering behaviors

of the Draco lizard [11]. Its suitability is assessed against PSO, WOA, and GWO using a PI that integrates four performance metrics: success rate (SR), convergence speed (CS), computational time (CPU time), and the standard deviation of search agent positions (STD). Because these algorithms are stochastic in nature [12], their performance may vary between trials. Therefore, 1,000 independent trials were conducted for each algorithm under each PSC, and the resulting average performance values were used to calculate the final PI scores. The highest-ranked algorithm was proposed as a promising candidate for future MPPT implementation under partial shading conditions.

The main contributions of this study are as follows: 1) the formulation of a PI that integrates four weighted metrics relevant to MPPT performance; 2) a statistically robust evaluation based on 1,000 independent trials; and 3) a comparative assessment against benchmark algorithms under three PSCs representing different GMPP locations. The remainder of this study is organized as follows. Section II presents the PSC formulation. Section III describes DLO and the benchmark algorithms. Section IV introduces the PI formulation. Section V presents and discusses the simulation results. Finally, Section VI concludes the study.

II. Partial Shading Conditions

This section presents the PV array model and the irradiance patterns used to define the partial shading test cases. These conditions determine the corresponding GMPP locations used in the subsequent algorithm evaluation. A solar cell is the fundamental building block of a PV array. In this study, the two-diode model is adopted because it offers higher modeling accuracy than the single-diode model while maintaining lower complexity than the three-diode model [10, 13, 14]. The equivalent circuit and the corresponding solar-cell output-current equation, I_{cell}, are presented in Fig. 1 and Eq. (1), respectively.

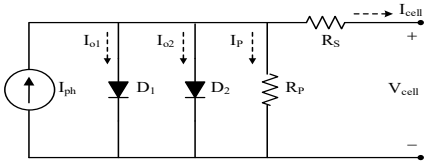


Fig. 1. The two-diode PV cell model.

$$I_{cell} = I_{ph} - I_{o1}\left[\exp\left(\frac{V_{cell} + I_{cell}R_s}{a_1 V_{T1}}\right) - 1\right] - I_{o2}\left[\exp\left(\frac{V_{cell}+I_{cell}R_s}{a_2 V_{T2}}\right) - 1\right] - \left(\frac{V_{cell}+I_{cell}R_s}{R_P}\right) \quad (1)$$

where I_{ph} is the photocurrent; I_{o1} and I_{o2} are the reverse saturation currents of diodes D_1 and D_2, respectively; V_{T1} and V_{T2} are the corresponding thermal voltages; a_1 and a_2 are the diode ideality factors; V_{cell} is the cell output voltage; and Rs and R_p are the series and parallel resistances, respectively.

A PV module is formed by connecting solar cells in series and parallel. This study uses the 50-W Solar Power Mart SPM050-P module, which consists of 36 series-connected cells. Its electrical parameters under standard test conditions, G = 1000 W/m² and T = 25 °C, are listed in Table I. The PV array consists of one string containing five series-connected modules, forming a 5S1P configuration, as shown in Fig. 2. When all modules are exposed to identical irradiance and temperature, the PV array exhibits only one maximum power point (MPP).

TABLE I. Parameters of the PV Module of SPM050-P at STC

Parameters	Values
Maximum Power (P_{MAX})	50 W_P
Voltage at Pmax (V_{MPP})	18.00 V
Current at Pmax (I_{MPP})	2.78 A
Open circuit voltage (V_{OC})	21.80 V
Short circuit current (I_{SC})	2.97 A
Temperature Coefficients of P_{MAX}	-0.45%/°C
Temperature Coefficients of V_{OC}	-0.35%/°C
Temperature Coefficients of I_{SC}	0.05%/°C
NOCT	47 °C

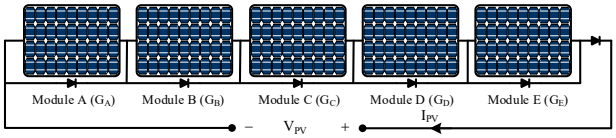


Fig. 2. PV array with 1 parallel string of 5 modules in series.

The P–V and I–V characteristics under various irradiance (G) and temperature (T) levels are presented in Fig. 3 and Fig. 4, respectively.

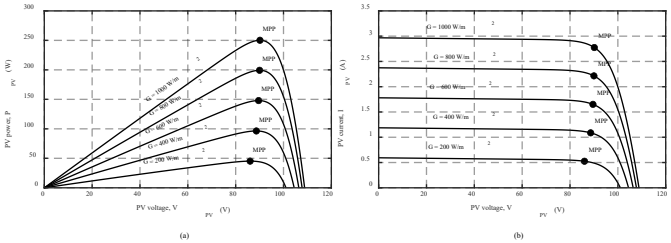


Fig. 3. (a) P–V and (b) I–V curves for various values of G (G= 200 W/m^2 to G= 1000 W/m^2).

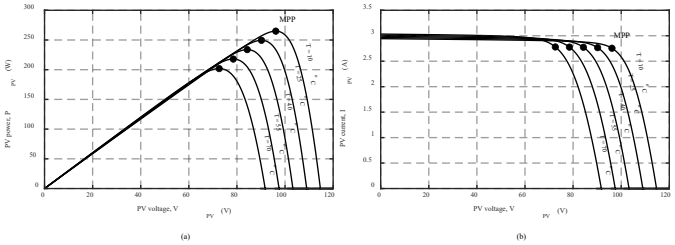


Fig. 4. (a) P–V and (b) I–V curves for various values of T (T= 10 °C to T= 70 °C).

Changes in irradiance primarily influence the PV current and output power, while temperature variations mainly affect the PV voltage. Accordingly, the temperature is maintained at 25 °C throughout this study to focus the analysis on irradiance

effects. Under partial shading, unequal irradiance across the modules produces several LMPPs and a single GMPP. Three representative cases are examined, as shown in Fig. 5 and listed in Table II: PSC1 with the GMPP on the left, PSC2 with the GMPP near the center, and PSC3 with the GMPP on the right.

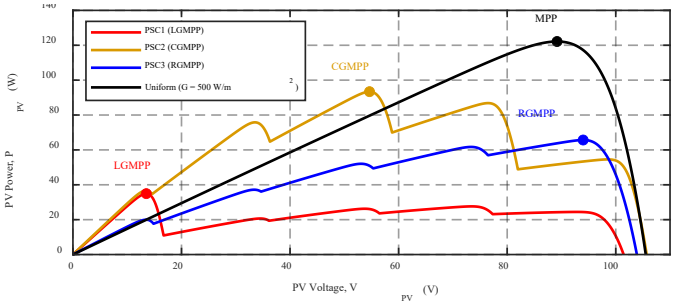


Fig. 5. P–V curves of the PV array for PSC1 (LGMPP), PSC2 (CGMPP), PSC3 (RGMPP) and uniform G = 500 W/m².

TABLE II. IRRADIANCE PARAMETERS OF THE PV ARRAY IN FIG. 5

PSC Pattern	Module Irradiance (G =1.0 = 1000 W/m²)					GMPP	
	G_A	G_B	G_C	G_D	G_E	V_{MPP} (V)	P_{MPP} (W)
Uniform	0.50	0.50	0.50	0.50	0.50	89.19	122.15
1: LGMPP	0.10	0.14	0.18	0.22	0.95	13.52	34.92
2: CGMPP	0.20	0.40	0.60	0.80	1.00	54.64	93.40
3: RGMPP	0.25	0.30	0.35	0.40	0.55	93.99	65.67

In this study, each evaluated algorithm used a lookup table to calculate the fitness values under the three PSC scenarios shown in Fig. 5, namely PSC1, PSC2, and PSC3. The lookup table inputs, V_{PV} and I_{PV}, were generated using the MATLAB/Simulink PV system simulator developed in [15]. The resulting dataset contains 10,000 distinct V_{PV}–I_{PV} data pairs, a sample of which is presented in Fig. 6.

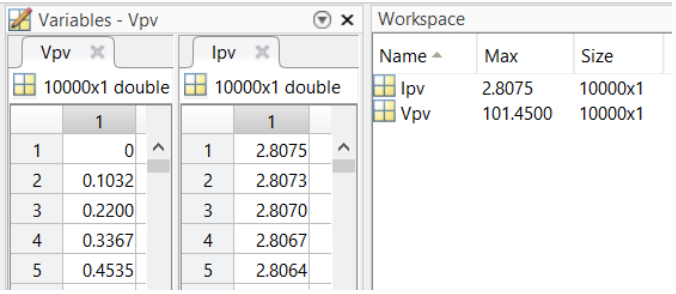


Fig. 6. Sample V_{PV} and I_{PV} data used in the lookup table for fitness evaluation.

III. DRACO LIZARD OPTIMIZER

The Draco Lizard Optimizer (DLO) [11] is a population-based metaheuristic inspired by the gliding, navigation, camouflage, and adaptive maneuvering behaviors of the Draco lizard. Similar to most metaheuristic algorithms, its optimization process consists of four main phases: initialization, update position, selection, and evaluation of the stopping criterion, as illustrated in Fig. 7.

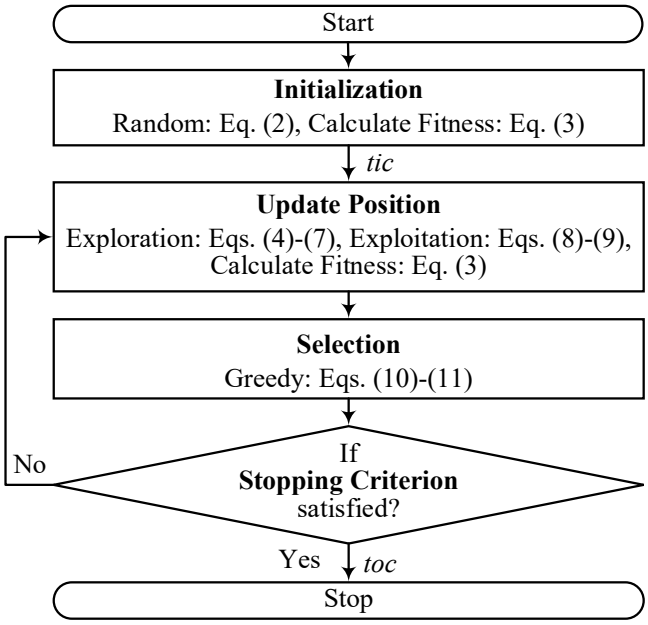


Fig. 7. The optimization process of DLO.

A. Initialization

As in most metaheuristic algorithms, the initial population of Draco lizards consists of N search agents, each located in a D-dimensional search space. The search agents are randomly generated within the predefined search boundaries using Eq. (2).

$$X_i^{(j)} = Lb^{(j)} + rand \cdot \left(Ub^{(j)} - Lb^{(j)}\right), \quad i = 1,2,\dots,N;\ j = 1,2,\dots,D \tag{2}$$

where *rand* is a uniformly distributed random number in the interval [0,1]. The terms $Lb^{(j)}$ and $Ub^{(j)}$ denote the lower and upper bounds of the j[th] dimension, respectively.

In this study, the lower search boundary was fixed at 0 V, while the upper boundary was set to the corresponding open-circuit voltage (V_{oc}) of each partial shading condition. The position of the i[th] search agent in the j[th] dimension is represented as $X_i^{(j)}$, constrained by $Lb^{(j)} \le X_i^{(j)} \le Ub^{(j)}$. Any search agent position exceeding the predefined boundaries was clipped to the corresponding lower or upper boundary before fitness evaluation. The fitness of each search agent was evaluated using partial shading condition lookup tables containing 10,000 (V_{PV}, I_{PV}) data pairs. Linear interpolation was employed to estimate the photovoltaic current when the search agent voltage did not exactly match a stored value. The corresponding photovoltaic power was then calculated and used as the fitness value. For global maximum power point identification, the optimization objective was to maximize the photovoltaic array output power, as expressed in Eq. (3).

$$\max f(x) = P_{PV}(x) = V_{PV}(x) \times I_{PV}(x) \quad (3)$$

where V_{PV}, I_{PV} and P_{PV} denote the PV array output voltage, current, and power, respectively.

B. Update position

Phase 1: Exploration (Gliding and Navigation)

During the first half of the optimization process, gliding and directional navigation are used to promote global exploration. The Draco lizard possesses strong environmental awareness and visual navigation abilities, allowing it to select suitable flight paths, control its posture, avoid obstacles, and land accurately after gliding over long distances. When $t < 0.5 \cdot T_{max}$, DLO enters the exploration phase. During this stage, the algorithm imitates the Draco lizard's ability to control its gliding direction and landing position. The search agents update their positions based on their historical best solutions, randomly selected agents, and a directional turning vector. This mechanism enables the population to explore different regions of the search space while progressively approaching promising solutions. The update position strategy is expressed in Eq. (4).

$$\overrightarrow{x_i(t+1)} = \begin{cases} \overrightarrow{xbest_i} + r \cdot \left(\overrightarrow{x_m(t)} - A \cdot \overrightarrow{xbest_i}\right) + \\ \overrightarrow{DL} \cdot r \cdot \left(\overrightarrow{x_m(t)} - \overrightarrow{x_n(t)}\right), \text{ if } f\left(\overrightarrow{xbest_i}\right) > f\left(\overrightarrow{xbest_m}\right) \\ \overrightarrow{xbest_i} + r \cdot \left(\overrightarrow{xbest_i} - \overrightarrow{x_m(t)}\right) + \\ \overrightarrow{DL} \cdot r \cdot \left(\overrightarrow{x_m(t)} - \overrightarrow{x_n(t)}\right), \text{ otherwise} \end{cases} \quad (4)$$

where, $\overrightarrow{xbest_i}$ denotes the historical best position of the i^{th} search agent. The vector r is a D-dimensional random vector whose elements are uniformly distributed within [0, 1]. The vectors $\overrightarrow{x_m(t)}$ and $\overrightarrow{x_n(t)}$ represent the positions of two distinct Draco lizards randomly selected from the population at iteration t. The term f denotes the objective function, while t and $t+1$ represent the current and subsequent iterations, respectively. The random coefficient A is generated using Eq. (5).

$$A = round(1 + rand) \quad (5)$$

where A takes a value of either 1 or 2, and *round()* denotes the rounding function. The turning direction of the Draco lizard during gliding is represented by the vector defined in Eq. (6).

$$\overrightarrow{DL} = \left[DL^{(1)}, DL^{(2)}, DL^{(3)}, \ldots, DL^{(D)}\right] \quad (6)$$

where each component is defined using Eq. (7).

$$DL^{(j)} = \begin{cases} 1, & \text{if } t < \varphi_1 T_{\max}, \\ P_1(k), & \text{else if } t < \varphi_2 T_{\max}, \\ P_2(k), & \text{else if } t < \varphi_3 T_{\max}, \\ P_3(k), & \text{else if } t < \varphi_4 T_{\max}, \\ 0, & \text{otherwise}, \end{cases} \quad j = 1,2,\ldots,D \quad (7)$$

where k is a randomly generated integer within [1, 4]. The threshold values are $\varphi_1 = 0.1$, $\varphi_2 = 0.2$, $\varphi_3 = 0.3$, and $\varphi_4 = 0.4$. The predefined vectors are $P_1 = [0, 1, 1, 1]$, $P_2 = [0, 0, 1, 1]$, and $P_3 = [0, 0, 0, 1]$. Thus, $\overrightarrow{DL}$ is a D-dimensional binary vector composed of zeros and ones. Since GMPP identification is formulated as a one-dimensional voltage-search problem, one component of the generated directional vector was used as the scalar directional coefficient in the position-update equation. The maneuverable and flexible gliding strategy employed by the Draco lizard during the exploration phase is illustrated in Fig. 8. When avoiding predators such as snakes, the Draco lizard adjusts its flight path according to the surrounding environment. The first differential term, $\left(\overrightarrow{xbest_i} - \overrightarrow{x_m(t)}\right)$, enables the lizard to move from its historical best position toward a distant tree trunk. The second differential term, $\left(\overrightarrow{x_m(t)} - \overrightarrow{x_n(t)}\right)$, allows it to respond rapidly to environmental cues and move toward a safer region. Meanwhile, the directional vector $\overrightarrow{DL}$ provides the flexibility needed to change the gliding direction. Together, these mechanisms enhance population diversity and support effective global exploration of the search space.

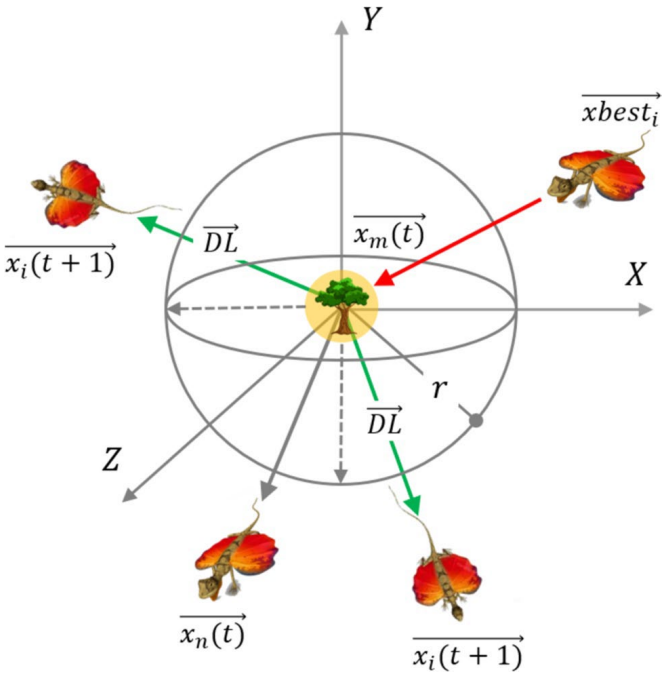


Fig. 8. Flexible gliding strategy of the Draco lizard during the exploration phase.

Phase 2: Exploitation (Camouflage and Maneuvering)

During the second half of the optimization process, DLO enters the exploitation phase, defined by $t \geq 0.5 \cdot T_{max}$. This phase is inspired by the Draco lizard's ability to respond rapidly to threats by either remaining motionless for camouflage or escaping through sudden maneuvering. As the iterations progress, the population becomes more concentrated around promising regions of the search space. To refine the search while reducing the risk of entrapment at local optima, DLO combines adaptive Gaussian mutation with occasional Lévy flights. The Gaussian mutation mechanism promotes local exploitation by directing each search agent toward the global best solution while progressively reducing the search range. In contrast, the Lévy flight mechanism introduces occasional long-distance movements, enabling the population to escape

local optima and explore distant regions. A switching probability p determines which strategy is applied at each iteration. The position update during this phase is expressed in Eq. (8).

$$\overrightarrow{x_i(t+1)} = \begin{cases} \overrightarrow{gbest} + levy(D) \cdot \left(\overrightarrow{gbest} - \overrightarrow{x_i(t)}\right), & \text{if } rand(0,1) < p, \\ \overrightarrow{gbest} + g \cdot \lambda \cdot \left(\overrightarrow{xbest_i} - \overrightarrow{x_i(t)}\right), & \text{otherwise} \end{cases} \quad (8)$$

where $p = 0.2$, t is the current iteration, and T_{max} is the maximum number of iterations. The vector $\overrightarrow{gbest}$ denotes the best position found by the population, while $\overrightarrow{xbest_i}$ is the historical best position of the i[th] search agent. The parameter g is a normally distributed random variable with zero mean and a variance of 10. The factor $\lambda = \left(1 - \frac{t}{T_{max}}\right)$ decreases linearly from 0.5 at the beginning of the exploitation phase to 0 at the final iteration, progressively reducing the Gaussian mutation step and enabling finer local search around the global best solution. The Lévy flight step is calculated using Eq. (9).

$$levy = 0.05 \cdot \frac{u \cdot \rho}{|v|^{\frac{1}{\beta}}}, \rho = \left(\frac{\Gamma(1+\beta) \cdot \sin\left(\frac{\pi\beta/2}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \cdot \beta \cdot 2^{(\beta-1)/2}}\right)^{1/\beta} \quad (9)$$

with $\beta = 1.5$. The variables u and v are random variables used to generate the Lévy step, and $\Gamma()$ denotes the gamma function.

The camouflage and adaptive maneuvering strategies used during the exploitation phase are illustrated in Fig. 9. Gaussian mutation gradually narrows the search region around the global best solution, allowing the search agents to perform a more detailed local search. Meanwhile, occasional Lévy flights enable the agents to move beyond the current region and avoid entrapment at local optima. The interaction between these two mechanisms improves the convergence stability and exploitation capability of DLO.

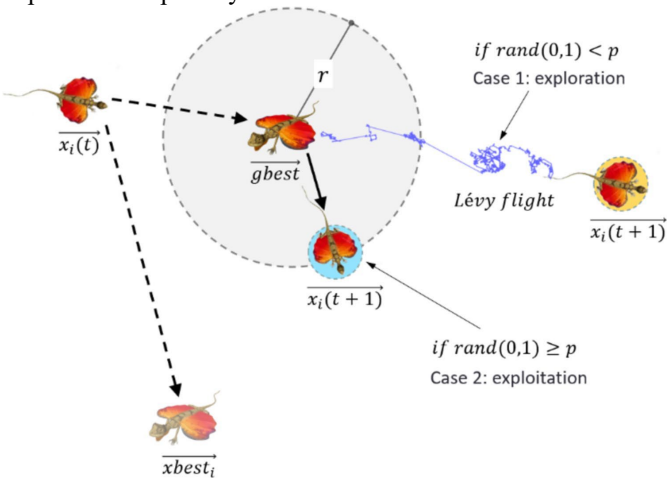


Fig. 9. Camouflage and adaptive maneuvering strategies of the Draco lizard during the exploitation phase.

C. Selection

After each iteration, DLO applies a greedy selection mechanism to update the individual best position of each search agent and the global best position of the population. A newly generated position is retained only when it produces a better fitness value than the previously recorded solution. For the maximization problem considered in this study, the individual and global best positions are updated using Eqs. (10) and (11), respectively:

$$\overrightarrow{xbest_i} = \begin{cases} \overrightarrow{x_i(t+1)}, & \text{if } f\left(\overrightarrow{x_i(t+1)}\right) > f\left(\overrightarrow{xbest_i}\right), \\ \overrightarrow{xbest_i}, & \text{otherwise.} \end{cases} \quad (10)$$

$$\overrightarrow{gbest} = \begin{cases} \overrightarrow{x_i(t+1)}, & \text{if } f\left(\overrightarrow{x_i(t+1)}\right) > f\left(\overrightarrow{gbest}\right), \\ \overrightarrow{gbest}, & \text{otherwise.} \end{cases} \quad (11)$$

where $\overrightarrow{x_i(t+1)}$ is the i[th] Draco lizard position vector generated during the current iteration, and *(t + 1)* is the current iteration number.

D. Stopping Criterion

The update position and selection processes are repeated until the stopping criterion is satisfied. In this study, the algorithm terminates when the maximum number of iterations, T_{max} = 100, is reached. Due to the stochastic components inherent in most metaheuristic algorithms, including the benchmark algorithms used in this study and DLO, as reflected in Eqs. (2)–(9), their performance may vary across independent trials, particularly when a small population size is used. An unfavorable initial population may cause the search agents to converge prematurely to an LMPP rather than the GMPP. Increasing the population size may improve the probability of locating the GMPP, but it also increases the convergence time and computational burden. Therefore, 1,000 independent trials were conducted for each test to account for algorithmic randomness and provide a statistically reliable performance assessment. The reported SR, CS, CPU time, and STD values were subsequently calculated from the corresponding trial results.

E. Selected Benchmark Algorithms: PSO, WOA, and GWO

In this study, PSO, WOA, and GWO were selected as benchmark algorithms for comparison with DLO. PSO [6] is inspired by the collective movement and social behavior of bird flocks and fish schools, in which each particle updates its position based on its personal best experience and the global best solution identified by the population. Its simple structure and rapid convergence make it effective for continuous optimization problems. WOA [7] is inspired by the bubble-net hunting behavior of humpback whales and performs the search through random exploration, encircling, and spiral movements. These mechanisms enable WOA to explore multimodal search spaces while reducing the likelihood of entrapment at local optima. GWO [8] imitates the leadership hierarchy and cooperative hunting behavior of grey wolves, in which the three best solutions, designated as alpha, beta, and delta, guide the remaining search agents toward promising regions of the search space. This hierarchical mechanism provides a systematic balance between exploration and exploitation.

These algorithms were selected because they are among the most popular and well-established population-based metaheuristics, have demonstrated effectiveness in solving

complex optimization problems, have been widely adopted in photovoltaic MPPT applications, and employ distinct search philosophies and exploration–exploitation mechanisms suitable for the nonlinear and multimodal search spaces associated with partial shading conditions. The parameter settings used for all algorithms are summarized in Table III. All evaluated algorithms used the same population size, maximum number of iterations, search boundaries, initialization method, and stopping criterion. Each search agent also underwent one objective-function evaluation per iteration, ensuring a fair performance comparison across all algorithms. The MATLAB source codes used in this study are publicly available through MATLAB Central File Exchange.

TABLE III. PARAMETERS OF THE DLO, PSO, WOA AND GWO

Parameters	Value
Population size, N	5
Lower boundary, LB	0
Upper boundary, UB	Voc of PSC1, PSC2 & PSC3
Initialization	Randomize
Convergence control parameters	
DLO	Linearly decrease: $\lambda = 0.5 \rightarrow 0$
PSO	Linearly decrease: $\omega = 0.7 \rightarrow 0.4$
WOA	Linearly decrease: $a = 2 \rightarrow 0$, $a2 = -1 \rightarrow -2$
GWO	Linearly decrease: $a = 2 \rightarrow 0$
Maximum iteration	100
Maximum trial	1,000

IV. PERFORMANCE INDEX

In this study, a performance index (PI) is introduced to identify the most suitable algorithm among the evaluated metaheuristic methods for potential MPPT applications. The PI integrates four weighted performance metrics: success rate (SR), convergence speed (CS), CPU time, and search agent standard deviation (STD), as defined in Eq. (12):

$$PI^{i}_{ave} = \left(k_1 \cdot \frac{SR^{i}}{SR_{max}}\right) + \left(k_2 \cdot \frac{CS_{min}}{CS^{i}}\right) + \left(k_3 \cdot \frac{CPU_{min}}{CPU^{i}}\right) + \left(k_4 \cdot \frac{STD_{min}}{STD^{i}}\right) \quad (12)$$

where SR^i, CS^i, CPU^i and STD^i represent the corresponding performance metrics of the i^{th} algorithm. SR_{max} denotes the maximum achievable success rate of 100%, while CS_{min}, CPU_{min}, and STD_{min} are the lowest corresponding values obtained among all evaluated algorithms. The weighting coefficients satisfy $k_1 + k_2 + k_3 + k_4 = 1$ and were assigned as $k_1 = 0.55$, $k_2 = 0.20$, $k_3 = 0.15$, and $k_4 = 0.10$ for SR, CS, CPU time, and STD, respectively. These coefficients were selected based on the authors' experience in metaheuristic-based MPPT research, with the greatest emphasis placed on SR, followed by CS, CPU time, and STD.

A sensitivity analysis was conducted by varying k_1 from 0.50 to 0.70, while k_2, k_3, and k_4 were proportionally adjusted to preserve their relative proportions and ensure that $k_1 + k_2 + k_3 + k_4 = 1$. Table IV summarizes the resulting PI values and algorithm rankings under the different weight combinations. DLO consistently remained ranked first, with its PI increasing from 0.9304 at k1 = 0.50 to 0.9488 at k1 = 0.70. Although the relative ranking of PSO and WOA changed at $k_1 = 0.70$, DLO retained the highest PI across all evaluated weight combinations. This confirms that the first-place ranking of DLO is robust within the investigated range of weighting coefficients.

TABLE IV. SENSITIVITY ANALYSIS OF PI WEIGHTING COEFFICIENTS

k_1	k_2	k_3	k_4	DLO PI (Rank)	PSO PI (Rank)	WOA PI (Rank)	GWO PI (Rank)
0.50	0.22	0.17	0.11	0.9304 (1)	0.9194 (2)	0.8606 (3)	0.8581 (4)
0.55	**0.20**	**0.15**	**0.10**	**0.9350 (1)**	**0.9137 (2)**	**0.8718 (3)**	**0.8650 (4)**
0.60	0.18	0.13	0.09	0.9396 (1)	0.9080 (2)	0.8830 (3)	0.8719 (4)
0.65	0.16	0.12	0.08	0.9442 (1)	0.9024 (2)	0.8941 (3)	0.8788 (4)
0.70	0.13	0.10	0.07	0.9488 (1)	0.8967 (3)	0.9053 (2)	0.8858 (4)

Note: Weighting coefficients are rounded to two decimal places for presentation, while PI values were calculated using the corresponding unrounded proportional weights.

A trial was classified as successful when the final best fitness reached at least 99.9% of the true global maximum power point, corresponding to a power tolerance not exceeding 0.1%. SR reflects an algorithm's reliability in locating the GMPP. A higher SR indicates more consistent GMPP identification and a lower likelihood of entrapment at an LMPP. CS is the first iteration at which the averaged best fitness reaches at least 99.9% of the true GMPP. Therefore, a lower CS indicates faster convergence and may reduce transient power loss in future MPPT implementation. CPU represents the computational time required to complete one iteration. A lower CPU time indicates a lower measured computational burden, which is generally associated with a simpler algorithmic structure and greater suitability for implementation on low-cost hardware. STD quantifies the distribution of the search agent positions around their mean at each iteration. A lower STD indicates smaller positional fluctuations and smoother convergence. In hardware implementation, this behavior may reduce duty-cycle oscillations and, consequently, thermal stress on the switching devices.

V. RESULTS AND DISCUSSION

All simulations were performed using MATLAB R2024a on a laptop equipped with an AMD Ryzen 9 4900H processor, 16 GB of RAM, and a 64-bit Windows 10 operating system. DLO, PSO, WOA, and GWO were independently evaluated over 1,000 trials under PSC1, PSC2, and PSC3. Figs. 10–13 present the averaged convergence trajectories over 100 iterations across all successful trials for DLO, PSO, WOA, and GWO, respectively, while Figs. 14–17 show the corresponding search agent distributions over 100 iterations for representative successful trials of the same algorithms.

As reported in Table V, DLO successfully located the GMPP in 935, 994, and 1,000 trials under PSC1, PSC2, and PSC3, corresponding to SR values of 93.50%, 99.40%, and 100.00%, respectively, and an average SR of 97.63%. In comparison,

PSO, WOA, and GWO achieved average SR values of 86.27%, 97.23%, and 92.73%, respectively. PSC1 was the most challenging condition because none of the algorithms achieved an SR of 100%, whereas all algorithms successfully located the GMPP under PSC3. The high SR of DLO demonstrates its robustness in multimodal search, particularly under PSC1, where the GMPP is located in the low-voltage region. Under this condition, the benchmark algorithms were more susceptible to premature convergence and entrapment at an LMPP. CS was evaluated from the averaged GMPP convergence curves of all successful trials under each PSC, as shown in Figs. 10–13. As reported in Table V, DLO reached within 0.1% of the true GMPP at iterations 53, 19, and 12 under PSC1, PSC2, and PSC3, respectively, yielding an average CS of 28.00 iterations. In comparison, PSO, WOA, and GWO recorded average CS values of 23.67, 31.67, and 30.33 iterations, respectively. The slower convergence under PSC1 indicates a more challenging search landscape than those under PSC2 and PSC3.

As depicted in Figs. 14–17, the search agents are widely dispersed during exploration and progressively concentrate around the GMPP during exploitation, indicating a gradual reduction in population diversity. At each iteration, the STD of the search agent positions was calculated using MATLAB's std(x) function. The resulting values were averaged over 100 iterations to obtain the STD for each trial and then averaged across all successful trials for each PSC. For the representative DLO trial shown in Fig. 14, the STD values were 5.9564, 7.5059, and 9.1660 V under PSC1, PSC2, and PSC3, respectively. Across all successful trials, the corresponding values were 9.6895, 4.9881, and 5.6336 V, yielding an average STD of 6.7704 V, as reported in Table V. By comparison, PSO, WOA, and GWO recorded average STD values of 5.3477, 10.4840, and 9.1314 V, respectively.

CPU time was measured using MATLAB's tic and toc functions within the main iteration loop, which includes the update position, selection, and stopping criterion evaluation stages shown in Fig. 7. As tabulated in Table V, the average CPU times per iteration recorded by DLO were 6.8289, 6.8583, and 6.7236 ms under PSC1, PSC2, and PSC3, respectively, resulting in an overall average of 6.8036 ms. In comparison, PSO, WOA, and GWO recorded average CPU times of 7.3306, 7.4737, and 7.2733 ms, respectively. Thus, DLO imposed the lowest measured computational burden among the evaluated algorithms.

Based on the weighted SR, CS, CPU time, and STD values evaluated using (12), the PI values of all algorithms were calculated and summarized in Table V. DLO achieved the highest PI of 0.9350 and ranked first, followed by PSO, WOA, and GWO with PI values of 0.9137, 0.8718, and 0.8650, respectively. Although DLO did not achieve the best CS or STD, its high SR and low CPU time resulted in the highest PI, demonstrating the most balanced overall performance in terms of GMPP identification reliability, convergence speed, computational efficiency, and search agent distribution.

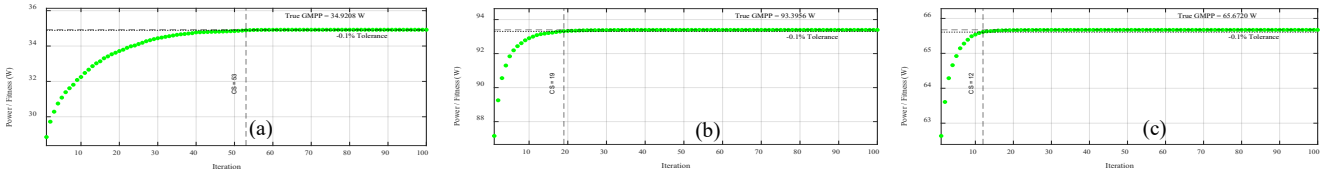


Fig. 10. Averaged convergence curves of DLO over all successful trials under: (a) PSC1, (b) PSC2, and (c) PSC3.

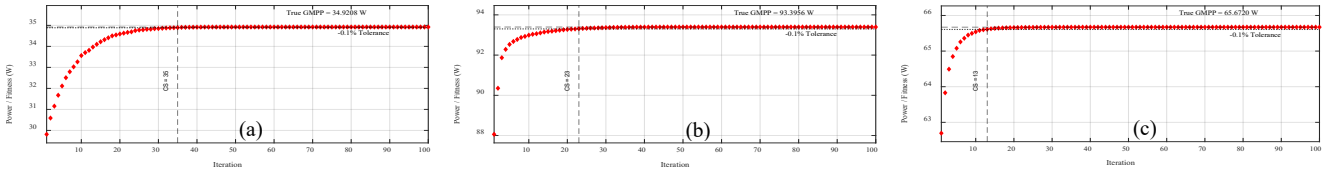


Fig. 11. Averaged convergence curves of PSO over all successful trials under: (a) PSC1, (b) PSC2, and (c) PSC3.

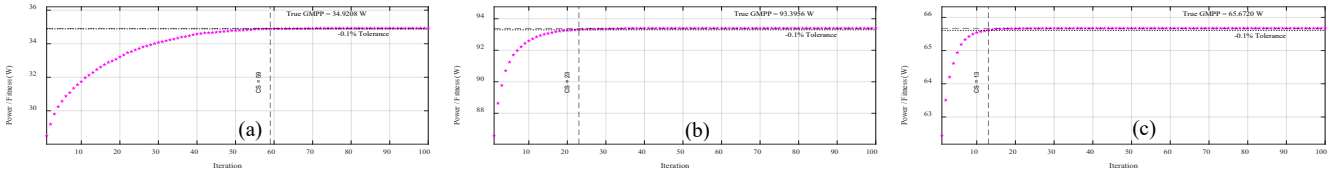


Fig. 12. Averaged convergence curves of WOA over all successful trials under: (a) PSC1, (b) PSC2, and (c) PSC3.

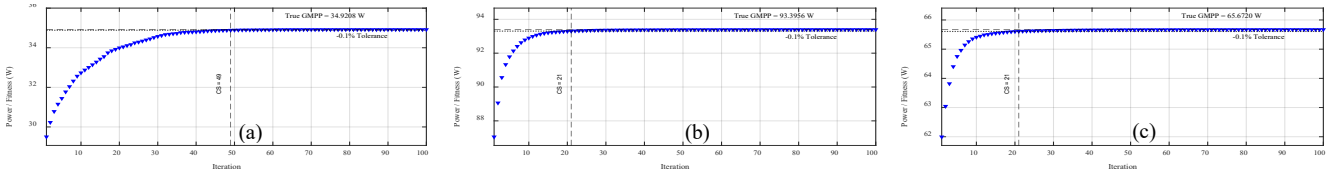


Fig. 13. Averaged convergence curves of GWO over all successful trials under: (a) PSC1, (b) PSC2, and (c) PSC3.

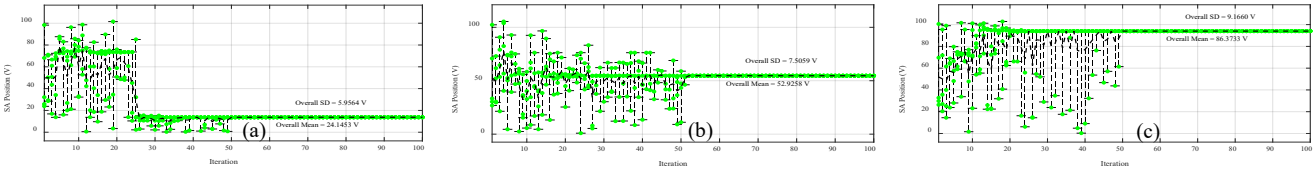


Fig. 14. Search agent distributions of DLO under: (a) PSC1, (b) PSC2, and (c) PSC3.

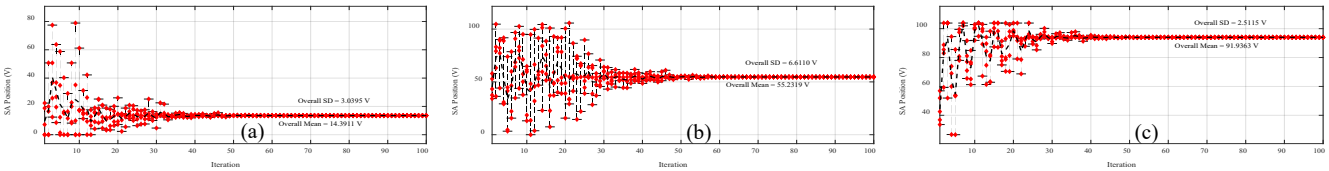


Fig. 15. Search agent distributions of PSO under: (a) PSC1, (b) PSC2, and (c) PSC3.

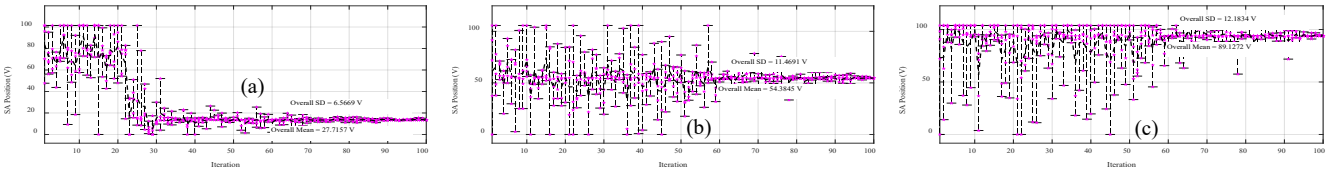


Fig. 16. Search agent distributions of WOA under: (a) PSC1, (b) PSC2, and (c) PSC3.

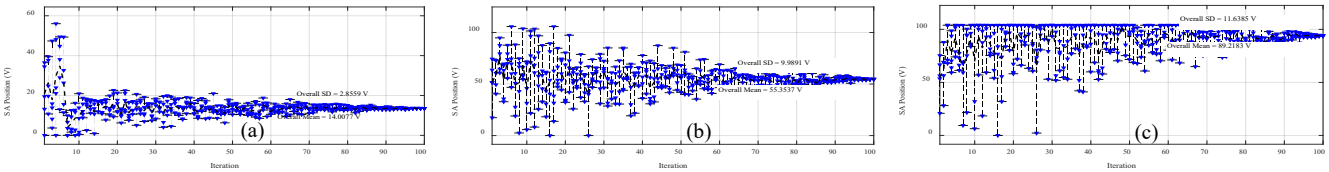


Fig. 17. Search agent distributions of GWO under: (a) PSC1, (b) PSC2, and (c) PSC3.

TABLE V. ALGORITHM PERFORMANCE SUMMARY OVER 1,000 TRIALS

Algo.	Case	SR (%)	CS (Iter.)	CPU (ms)	STD	PI (Rank)
DLO	PSC1	93.5	53	6.8289	9.6895	
	PSC2	99.4	19	6.8583	4.9881	
	PSC3	100.0	12	6.7236	5.6336	
	Ave.	**97.63**	**28.0**	**6.8036**	**6.7704**	**0.9350 (1)**
PSO	PSC1	58.8	35	7.3159	7.3881	
	PSC2	100.0	23	7.6478	5.8486	
	PSC3	100.0	13	7.0280	2.8064	
	Ave.	**86.27**	**23.67**	**7.3306**	**5.3477**	**0.9137 (2)**
WOA	PSC1	91.7	59	7.0901	5.8545	
	PSC2	100.0	23	7.3622	12.7243	
	PSC3	100.0	13	7.9689	12.8731	
	Ave.	**97.23**	**31.67**	**7.4737**	**10.4840**	**0.8718 (3)**
GWO	PSC1	78.2	49	7.0228	5.8864	
	PSC2	100.0	21	7.4291	9.8923	
	PSC3	100.0	21	7.3679	11.6155	
	Ave.	**92.73**	**30.33**	**7.2733**	**9.1314**	**0.8650 (4)**

VI. CONCLUSION

This study presented a PI-based evaluation of DLO for GMPP identification under PSCs as a preliminary assessment for future MPPT implementation. The evaluation was conducted as an offline numerical benchmarking study using pre-generated static PV lookup tables rather than as a complete converter-level MPPT implementation. DLO was benchmarked against PSO, WOA, and GWO through 1,000 independent trials under three PSC patterns representing left-, center-, and right-located GMPPs. The proposed PI integrated SR, CS, CPU time, and STD with respective weights of 0.55, 0.20, 0.15, and 0.10 to assess reliability, convergence performance, computational burden, and search agent dispersion. DLO achieved the highest average SR of 97.63% and the lowest average CPU time of 6.8036 ms per iteration. Although PSO recorded better average CS and STD values, DLO attained the highest PI of 0.9350, followed by PSO, WOA, and GWO with values of 0.9137, 0.8718, and 0.8650, respectively. These findings demonstrate that DLO provides the most balanced overall performance under various PSCs. Its individual metrics may be further improved through refined convergence-parameter tuning and more effective initialization strategies. However, because the fitness evaluation is based on static PV lookup tables, the present study does not account for real-time converter dynamics or closed-loop MPPT behavior. Future work will integrate DLO into a complete PV MPPT system incorporating power converter dynamics and validate its performance under

dynamic PSCs through converter-level simulation and hardware-level experimental testing.

VII. ACKNOWLEDGMENT

The authors would like to extend their sincere appreciation to members of the Faculty of Electrical Engineering, Universiti Teknologi MARA, Shah Alam, for their valuable support and assistance. Finally, the authors would like to thank all individuals who contributed, directly or indirectly, to the completion of this study.

REFERENCES

[1] N. Hashim, Z. Salam, D. Johari, and N. F. N. Ismail, "DC-DC boost converter design for fast and accurate MPPT algorithms in stand-alone photovoltaic system," *International Journal of Power Electronics and Drive Systems,* vol. 9, no. 3, p. 1038, 2018.

[2] N. Hashim and Z. Salam, "Critical evaluation of soft computing methods for maximum power point tracking algorithms of photovoltaic systems," *International Journal of Power Electronics and Drive Systems,* vol. 10, no. 1, p. 548, 2019.

[3] E. H. Houssein, M. A. Mahdy, D. Shebl, and W. M. Mohamed, "A survey of metaheuristic algorithms for solving optimization problems," in *Metaheuristics in machine learning: theory and applications*: Springer, 2021, pp. 515-543.

[4] N. Hashim, R. Baharom, M. M. Othman, D. Johari, and A. Ab. Rahman, "Hybrid iPSO-iADFHC MPPT algorithm for PV systems under various partial shading conditions," *Journal of Power Electronics,* vol. 26, no. 2, pp. 446-458, 2026.

[5] M. Jamhari, N. Hashim, M. M. Othman, and A. B. Abidin, "Metaheuristics-based maximum power point tracking for PV systems: a review," *International Journal of Power Electronics and Drive Systems (IJPEDS),* vol. 14, no. 4, p. 2495, 2023.

[6] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of ICNN'95-international conference on neural networks*, 1995, vol. 4: IEEE, pp. 1942-1948.

[7] S. Mirjalili and A. Lewis, "The Whale Optimization Algorithm," *Advances in Engineering Software,* vol. 95, pp. 51-67, May 2016, doi: 10.1016/j.advengsoft.2016.01.008.

[8] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey Wolf Optimizer," *Advances in Engineering Software,* vol. 69, pp. 46-61, Mar 2014, doi: 10.1016/j.advengsoft.2013.12.007.

[9] S. P. Adam, S. A. N. Alexandropoulos, P. M. Pardalos, and M. N. Vrahatis, "No Free Lunch Theorem: A Review," in *Conference on Approximation and Optimization*, Athens, GREECE, Jun 29-30 2017, vol. 145, in Springer Optimization and Its Applications, 2019, pp. 57-82, doi: 10.1007/978-3-030-12767-1_5. [Online]. Available: <Go to ISI>://WOS:000493065200005

[10] N. Hashim, R. Baharom, A. F. Abidin, K. S. Muhammad, and W. N. W. A. Munim, "Snow Ablation Optimizer-Based MPPT for PV Systems Under Various Partial Shading Conditions," in *2025 IEEE Industrial Electronics and Applications Conference (IEACon)*, 2025: IEEE, pp. 98-103.

[11] X. Wang, "Draco lizard optimizer: a novel metaheuristic algorithm for global optimization problems," *Evolutionary Intelligence,* vol. 18, no. 1, p. 10, 2025.

[12] N. Hashim, N. F. N. Ismail, D. Johari, I. Musirin, and A. A. Rahman, "Optimal population size of particle swarm optimization for photovoltaic systems under partial shading condition," *Int. J. Elec. & Comp. Eng,* vol. 12, pp. 4599-4613, 2022.

[13] N. Hashim, M. K. A. M. Jamhari, and R. Baharom, "Improved particle swarm optimization MPPT for standalone PV system under varying environmental conditions," in *2024 IEEE 4th international conference in power engineering applications (ICPEA)*, 2024: IEEE, pp. 19-24.

[14] M. Jamhari, N. Hashim, R. Baharom, and M. M. Othman, "Simulation and verification of improved particle swarm optimization for maximum power point tracking in photovoltaic systems under dynamic environmental conditions," *International Journal of Power Electronics and Drive Systems (IJPEDS),* vol. 16, no. 1, pp. 608-621, 2025.

[15] K. Ishaque and Z. Salam, "A comprehensive MATLAB Simulink PV system simulator with partial shading capability based on two-diode model," *Solar energy,* vol. 85, no. 9, pp. 2217-2227, 2011.