

Optimal population size of particle swarm optimization for photovoltaic systems under partial shading condition

Norazlan Hashim¹, Nik Fasdi Nik Ismail¹, Dalina Johari¹, Ismail Musirin¹, Azhan Ab. Rahman²

¹School of Electrical Engineering, College of Engineering, Universiti Teknologi MARA, Shah Alam, Malaysia

²Faculty of Electrical Engineering, Universiti Teknikal Malaysia Melaka, Durian Tunggal, Malaysia

Article Info

Article history:

Received Jul 19, 2021

Revised May 31, 2022

Accepted Jun 12, 2022

Keywords:

Optimal population size

Partial shading condition

Particle swarm optimization

Performance index

ABSTRACT

Particle swarm optimization (PSO) is the most widely used soft computing algorithm in photovoltaic systems to address partial shading conditions (PSC). The success of the PSO run heavily depends on the initial population size (NP). A higher NP increases the probability of a global peak (GP) solution, but at the expense of a longer convergence time. To find the optimal value of NP, a trade-off is typically made between a high success rate and a reasonable convergence time. The most used trade-off method is a trial-and-error approach that lacks explicit guidelines and empirical evidence from detailed analysis, which can affect data reproducibility when different systems are used. Hence, this study proposes an empirical trade-off method based on the performance index (PI) indicator, which takes into account the weighted importance of success rate and convergence time. Furthermore, the impact of NP on achieving a successful PSO was empirically investigated, with the PSO tested with 16 different NPs ranging from 3 to 50, and 10,000 independent runs on various PSC problems. Overall, this study found that the best NP to use was 25, which had the best average PI value of 0.9373 for solving all PSC problems under consideration.

This is an open access article under the [CC BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



Corresponding Author:

Norazlan Hashim

School of Electrical Engineering, College of Engineering, Universiti Teknologi MARA

Shah Alam 40450, Selangor, Malaysia

Email: azlan4477@uitm.edu.my

1. INTRODUCTION

Recently, photovoltaic (PV) system is emerging as one of the most popular sources of renewable energy owing to its clean and inexhaustible nature [1]–[3]. Generally, the PV system is composed of a PV array, an intermediate DC/DC converter, and a DC/AC converter, which allows the flow of power in either grid-connected or stand-alone application. Generally, the performance of the PV system is negatively influenced by the occurrence of partial shading conditions (PSCs). Partial shading happens when different PV modules are subjected to different irradiance levels due to shading by buildings, trees, chimneys, dust, clouds or bird droppings [4]. Consequently, the power–voltage (P–V) characteristics curves exhibited multiple peaks with several local peaks (LPs) and one global peak (GP) to operate in maximum power point (MPP) of the PV system [3], [5]. As a result, these systems may produce lower power than the optimal operating point due to mismatch loss [6]. To mitigate this issue, many soft computing (SC)-based maximum power point tracking (MPPT) algorithms have been employed to drive the operating point towards the MPP on the P–V curve. The SC exploits the tolerance for imprecision, partial truth and uncertainty to achieve approximate, robust and low-cost optimal solutions [7], [8]. Since the SC algorithm search includes all the peaks over the entire P–V curve, it is possible to identify the GP.

Application of SC for MPPT has been extensively reviewed [9]–[13], which includes genetic algorithm, differential evolution, ant colony optimization, cuckoo search and particle swarm optimization. Each of these algorithms has specific and unique parameters which require adjustment to achieve the desired performance. A common parameter to all SCs is population size (NP). NP is the number of individuals/search agents/particles/chromosomes that exist in each generation/iteration of an algorithm and is one of the most important parameters which affect the achievement of GP solution. Traditionally, the NP is specified to be constant throughout the iteration process [14], [15]. At the initialization stage, the initial NP of search agents in the population known as a parent is randomly generated. Nevertheless, the random nature of the initialization stage causes uneven distribution of the initial population over the search space (SS). This lead the search towards unpromising regions (areas that do not contain GP solution) from the beginning [16]. It is recognized that large NP can ensure the diversity of search agents, hence improving the success rate of achieving GP solution. However, the larger the NP, the longer will be the sampling process runs, and the slower will be the convergence time. A previous study [17]–[20] suggested that NP should be in the range from 20 to 50 particles if there is no empirical study carried out to select the optimum NP. A good MPPT strategy should be able to successfully track the GP in a short time. Therefore, a trade-off is necessary between the success rate (SR) and the convergence time (CT) to attain the optimal value of NP. Commonly used trade-off methods include the trial and error method [21]–[23] which is time-consuming, the rule of thumb technique and the educated guess approach. These methods have no empirical guidelines or evidence from detailed performance analysis leading to challenges in reproducing experimental results, especially for different systems. Furthermore, limited previous studies have discussed the impact of NP on the successful performance of SC algorithms.

Hence, the present work proposes an empirical trade-off method to select the optimal value of NP. The proposed method is based on the performance index (PI) indicator which incorporates the weighted importance of SR and CT. Also, the impact of various NP values on the successful performance of the PSO algorithm was studied. Particle swarm optimization (PSO) was selected to implement the proposed method as PSO is the most widely used SC algorithm in solving various PSC problems. Besides, PSO is preferred due to its simple mathematical expressions and implementation [24], [25]. In the current work, PSO was tested with 16 different NPs of 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 35, 40, 45, and 50, on three types of PSC problems. The combination of the three problems could represent the actual case of no prior information about the location of the GP in the SS. GP locations can be categorized into three types of patterns: pattern 1, 2 and 3 with the GP located at the left side, middle and right side of the SS, respectively. To reduce the statistical errors, 10,000 independent runs of PSO were performed at each selected NP to solve each PSC problem.

2. RESEARCH METHOD

2.1. Problem formulation of partial shading

To study the impact of NP on the SR, PSO was applied to solve various PSC problems. The goal of optimization is to find the maximum value of fitness (P_{PVMax}) as determined using (1).

$$f_{max} = P_{PVMax} = \max\{V_{PV} \cdot I_{PV}\} \quad (1)$$

In (1), the V_{PV} and I_{PV} variables are the PV array output voltage and current, respectively. To simplify the evaluation of (1), the lookup table method was employed in this work. The P–V data for the lookup table was generated using a MATLAB/Simulink simulator that was developed in a previous study [26]. The simulator utilized a two-diode PV cell model which is superior to the single-diode model, particularly at low irradiance level [26]–[29] as depicted in Figure 1.

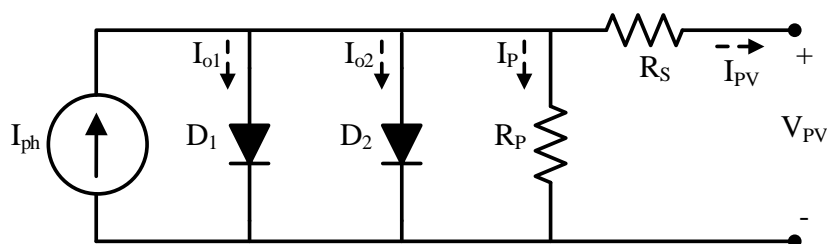


Figure 1. A two-diode model of a PV cell

For a module with N number of cells in series (N_{cell}), the output current of the module can be calculated:

$$I = I_{ph} - I_{o1} \left(\exp \left(\frac{V_{mpp} + I_{mpp} R_S}{a_1 V_{T1}} \right) - 1 \right) - I_{o2} \left(\exp \left(\frac{V_{mpp} + I_{mpp} R_S}{a_2 V_{T2}} \right) - 1 \right) - \left(\frac{V_{mpp} + I_{mpp} R_S \times N_{cell}}{I_{mpp} R_P \times N_{cell}} \right) \quad (2)$$

where I_{o1} and I_{o2} are the reverse saturation currents of diode 1 (D1) and diode 2 (D2), respectively; V_{T1} and V_{T2} are the thermal voltages of D1 and D2, respectively; a_1 and a_2 represent the diode ideality constants; V_{mpp} and I_{mpp} are the voltage and current of the PV cell at MPP, respectively. Several series-parallel connected PV modules can be set in the simulator to form the PV array with the desired voltage and current level. The input electrical parameters used for the simulator were based on the Solarex MSX-60 PV module [30] and the specifications at standard test conditions (STC) are displayed in Table 1.

Table 1. Electrical parameters of MSX-60 module at STC

Parameters	Values
Maximum Power (P_{max})	60 W
Voltage at Pmax (V_{mpp})	17.1 V
Current at Pmax (I_{mpp})	3.5 A
Open circuit voltage (V_{oc})	21.1 V
Short circuit current (I_{sc})	3.8 A
Temperature coeff. of V_{oc}	$-(80 \pm 10)$ mV/ $^{\circ}$ C
Temperature coeff. of I_{sc}	$-(0.065 \pm 0.015)\%$ / $^{\circ}$ C
Temperature coeff. of power	$-(0.5 \pm 0.05)\%$ / $^{\circ}$ C
NOCT	47 ± 2 $^{\circ}$ C
Operating Temperature	25 $^{\circ}$ C

In this work, the PV array was formed by two parallel five modules in series (5S2P) as illustrated in Figure 2. When partial shading occurs, different PV modules as illustrated in Figure 2 were subjected to different irradiance levels, which led to the appearance of multiple peaks in the $P-V$ characteristic curve. In reality, the location of GP in the SS ($P-V$ curves) is unknown. With no prior information provided, three PSC problems with different GP locations (pattern 1, 2 and 3 as mentioned earlier) in the SS were generated in this work as shown in Figure 3. The three patterns can be achieved when the PV modules as shown in Figure 2 are partially shaded with different values of irradiance as tabulated in Table 2.

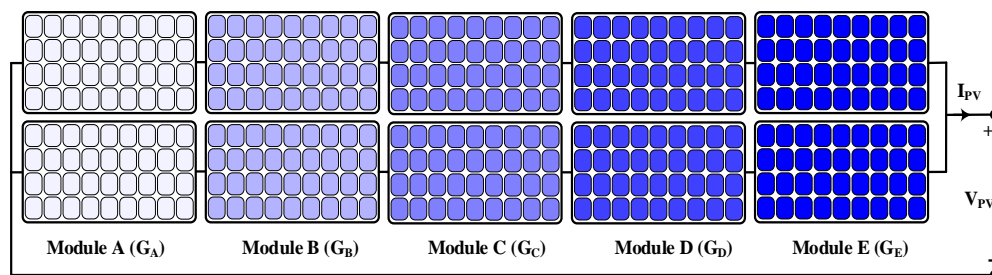


Figure 2. Schematic illustration of partially shaded PV arrays with varying irradiance patterns

The limit of SS for the PSC problems is in between the lower bound (LB) and upper bound (UB) of the PV array open-circuit voltage (V_{oc}). The LB was equal to 0 while the UB was equal to 105.65 V. For analysis purpose, the SS was divided into three sections, namely, the 1st section (from LB to $1/3 \times UB$), 2nd section (from $1/3 \times UB$ to $2/3 \times UB$), and 3rd section (from $2/3 \times UB$ to UB) as exhibited in Figure 3.

2.2. Particle swarm optimization

The PSO is a population-based SC algorithm developed by Eberhart and Kennedy [31] which was inspired by the social behavior and movement of bird flocking and fish schooling in the search of food sources. The search agent in the PSO is also known as a particle, while a group of particles is called a swarm or population. Each particle in the population carries a candidate solution for an optimization problem.

Iteratively, the particles explore the SS for an optimum solution by updating their position (x) and velocity (v). The global best position obtained by the population (G_{best}) and the best position reached by each particle (P_{best}) during movement was recorded at each i^{th} iteration. The movement of a particle in the SS is as illustrated in Figure 4 and was manipulated according to (4).

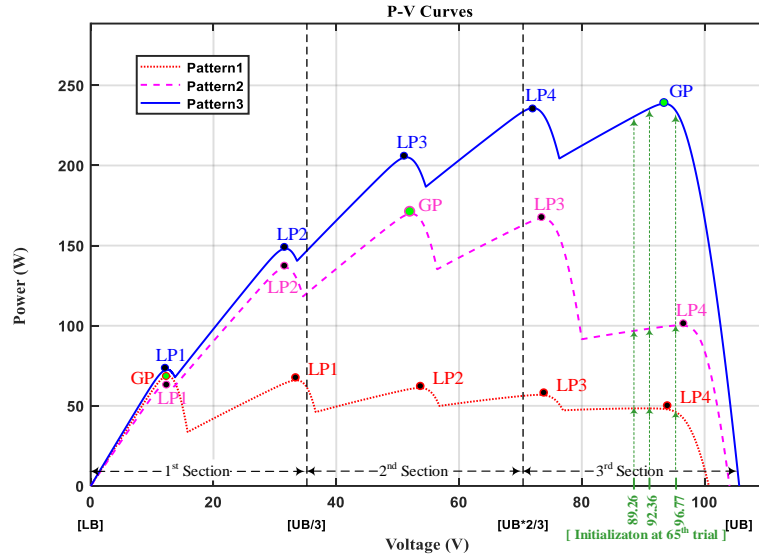


Figure 3. The power–voltage curve of PV arrays under various PSCs

Table 2. Module irradiance for shading patterns

PSC Pattern	Module Irradiance ($G=1.0=1000 \text{ W/m}^2$)					P_{PV} (W) and V_{PV} (V) at the Multipeak				
	G_A	G_B	G_C	G_D	G_E	LP1	LP2	LP3	LP4	GP
Pattern 1 (GP at 1 st section)	0.08	0.115	0.165	0.28	0.8	65.98 W 33.47 V	61.16 W 53.58 V	56.86 W 72.99 V	48.42 W 90.16 V	68.70 W 12.53 V
Pattern 2 (GP at 2 nd section)	0.15	0.315	0.45	0.6	0.75	64.69 W 12.59 V	136.24 W 31.70 V	166.91 W 73.57 V	100.23 W 95.99 V	169.90 W 52.23 V
Pattern 3 (GP at 3 rd section)	0.35	0.45	0.55	0.65	0.85	72.65 W 12.48 V	148.15 W 31.74 V	205.12 W 51.42 V	235.85 W 72.06 V	238.58 W 93.47 V

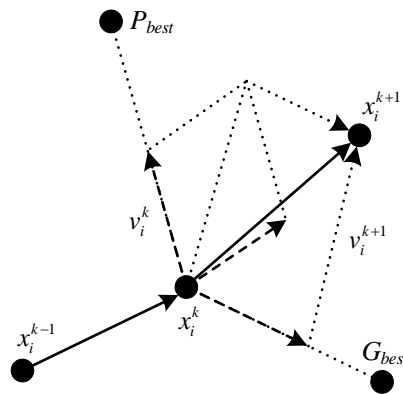


Figure 4. The movement of particles in PSO [32]

In the present study, to solve various PSC problems, benchmark stages of optimization process (initialization, reproduction, and selection) were implemented by PSO, as shown in Figure 5. In the initialization stage, the initial particles (x^{k-1}) were randomly generated within the boundaries of the SS as expressed in the (3).

```

for           i = 1:NP
   $x^{k-1}(i) = LB + (UB - LB) \cdot rand$ 
end

```

(3)

where NP is the number of particles in the population. The LB and UB of the SS were specified at 0 V and 105.65 V, respectively which were the range limit of the V_{oc} . Meanwhile, *rand* is the function that returns random numbers uniformly between 0 and 1.

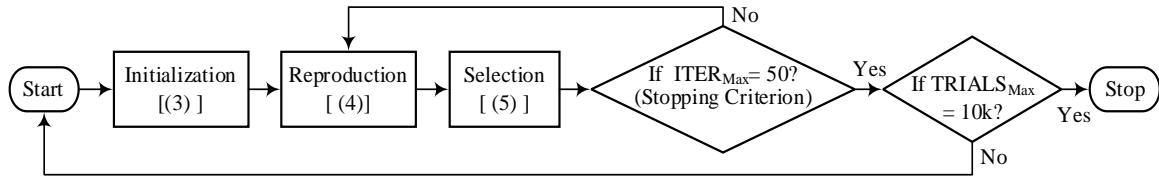


Figure 5. The benchmark optimization methodology for population-based SC algorithms [11]

The fitness of each initial particle was then evaluated and selected as the parent (x^k) population. To simplify, the lookup table method was used to evaluate the fitness (P_{PV}) of each particle. The initial G_{best} and P_{best} were also recorded during the initialization stage. In the reproduction stage of the i^{th} iteration, the offspring (x^{k+1}) was produced from the parent (x^k) population according to (4) based on a previous study [20], [33].

$$\begin{aligned}
 v_i^{k+1} &= w \cdot \{v_i^k + c_1 \cdot rand(P_{best} - x_i^k) + c_2 \cdot rand(G_{best} - x_i^k)\} \\
 x_i^{k+1} &= x_i^k + v_i^{k+1}
 \end{aligned}$$
(4)

where c_1 and c_2 are two acceleration constants regulating the relative velocity (v) regarding G_{best} and P_{best} and were selected such that $c_1 = c_2 = 1.49$; w is the inertia weight controlling the influence of the previous velocity on the current velocity. The w decreased linearly from 0.9 to 0.4 over the entire iterations. Meanwhile, the initial particle velocities (v^k) were randomly generated within 30% of the SS and were limited to the range during the run. All the parameter values applied were as proposed in previous studies [20], [34].

Next, in the selection stage, the best particles were selected to survive the next generation through a discriminatory process. In the selection stage, the highest potential particles within the population were selected while maintaining a constant NP for all the generations. In the present work, the competition selection method was selected due to its consistency (no randomness) which was proven to produce good results. The competition selection method enables both parent and offspring populations to compete with each other based on their corresponding fitness values. Then, the population with the best fitness value was selected as the winner to survive the next generation:

```

for           i = 1:NP
  if  $f(x_i^k) > f(x_i^{k+1})$ 
     $x_i^{k+1} = x_i^k$ 
  else  $x_i^{k+1} = x_i^{k+1}$ 
end

```

(5)

Finally, the reproduction and selection stages were repeated iteratively until a pre-defined stopping criterion is satisfied. The stopping criterion was used to terminate the algorithm. Typically, an algorithm is stopped after a specified maximum number of generations is reached or after the accuracy of final solutions reached a pre-defined threshold value. Termination could also happen once the best solution remained over a specified number of generations.

In the current study, the algorithm was stopped when the iteration reached the maximum number of 50 generations ($ITER_{Max}=50$). The $ITER_{Max}$ value of 50 is sufficient for the algorithm to reach the final solution of acceptable accuracy, wherein this study 0.1 W of the true GP was achieved. Generally, a higher maximum generation count produces more accurate final solutions despite the longer computational time. The entire simulation process was repeated by 10,000 independent trials ($TRIALS_{Max}=10,000$) to reduce the statistical errors that could arise from the uncertainty of randomness as outlined in (3) and (4).

2.3. Performance index analysis for optimal population size

The PI formulation as proposed in previous studies [35], [36] was adopted in this work to obtain the optimal NP for PSO in solving various PSC problems. For case study analysis, NP was varied in the sequence of 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 35, 40, 45 and 50. At each selected NP, PI values for all PSC problems that have been solved by PSO were calculated. The PI values were calculated based on the weighted importance (k_1 and k_2) of two important criteria in designing the MPPT which were the SR and the CT. In this study, the PSO successfully located the GP when the final solution was within the precision of 0.1 W of the true GP. In the MPPT application, the SR was influenced by the internal constants and variables of an algorithm such as NP, c_1 , c_2 , w , as expressed in (3) and (4). Meanwhile, external constant such as MPPT sampling time (T_{S_MPPT}) was included to determine the CT. The T_{S_MPPT} is the time given to the MPPT controller to read all the inputs and solve all the calculations involved in the algorithm. The T_{S_MPPT} is very much dependent on two main components of the converter circuit, the inductor and capacitor. For accurate MPPT, the T_{S_MPPT} must be specified to be greater than the settling time of transient response of a converter circuit. All the readings and calculations involved in the algorithm must be completed within the specified period. Otherwise, the algorithm might fail to perform in the required manner. However, large T_{S_MPPT} will reduce the speed of the CT and vice versa. In the present study, the T_{S_MPPT} was specified at 0.1 s, which was according to the previous study [37]. For further understanding, a simulation of PSO based MPPT with NP of 3 was carried out to solve pattern 1 (GP=68.70 W), and the MATLAB/Simulink model is as shown in Figure 6. An example of a successful GP tracking is illustrated in Figure 7. As observed after the initialization (Init.), the PSO successfully tracked the GP at the 5th iteration. Since NP=3, the input values (PV voltage, V_{PV} and current, I_{PV}) and calculations involved in the algorithm were completed within 0.3 s ($NP \times T_{S_MPPT}$) at each iteration/initialization. The CT was 1.8s as shown in Figure 7 which was calculated as $ITER \times NP \times T_{S_MPPT} = 6 \times 3 \times 0.1s = 1.8$ s. Hence, on average, to determine the CT of the i^{th} problem at different NP, the following general equation can be used:

$$CT^i = ITER_{Conv_Ave}^i \times NP \times T_{S_MPPT} \tag{6}$$

where $ITER_{CON_Ave}$ is the average number of iterations required to reach the near GP solution from the successful trials of the i^{th} problem.

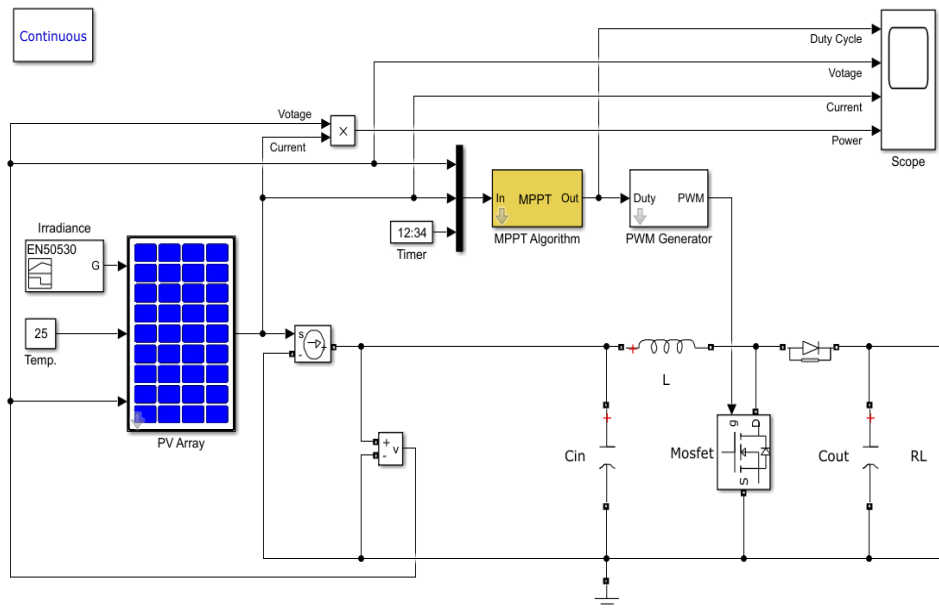


Figure 6. The MATLAB/Simulink model to simulate the PSO based MPPT

Meanwhile, at the selected NP, the PI of the i^{th} problem was calculated as (7):

$$PI^i = \left(k_1 \times \frac{SR^i}{N_T^i} \right) + \left(k_2 \times \frac{CT^i}{CT_{Min}^i} \right) \tag{7}$$

where the SR^i is the number of successful trials obtained using the PSO in solving the i^{th} problem, N_T is the total number of trials of the i^{th} problem (*i.e.* 10,000 runs), CT_{Min} refers to the minimum CT resulted from different NP in solving the i^{th} problem, k_1 and k_2 are the weighted importance applied to the SR and CT, respectively. The k_1 and k_2 are user-specified constants which satisfy $k_1+k_2=1$ based on the priority level of SR and CT. In the present study, k_1 and k_2 were specified at 90% (0.9) and 10% (0.1), respectively.

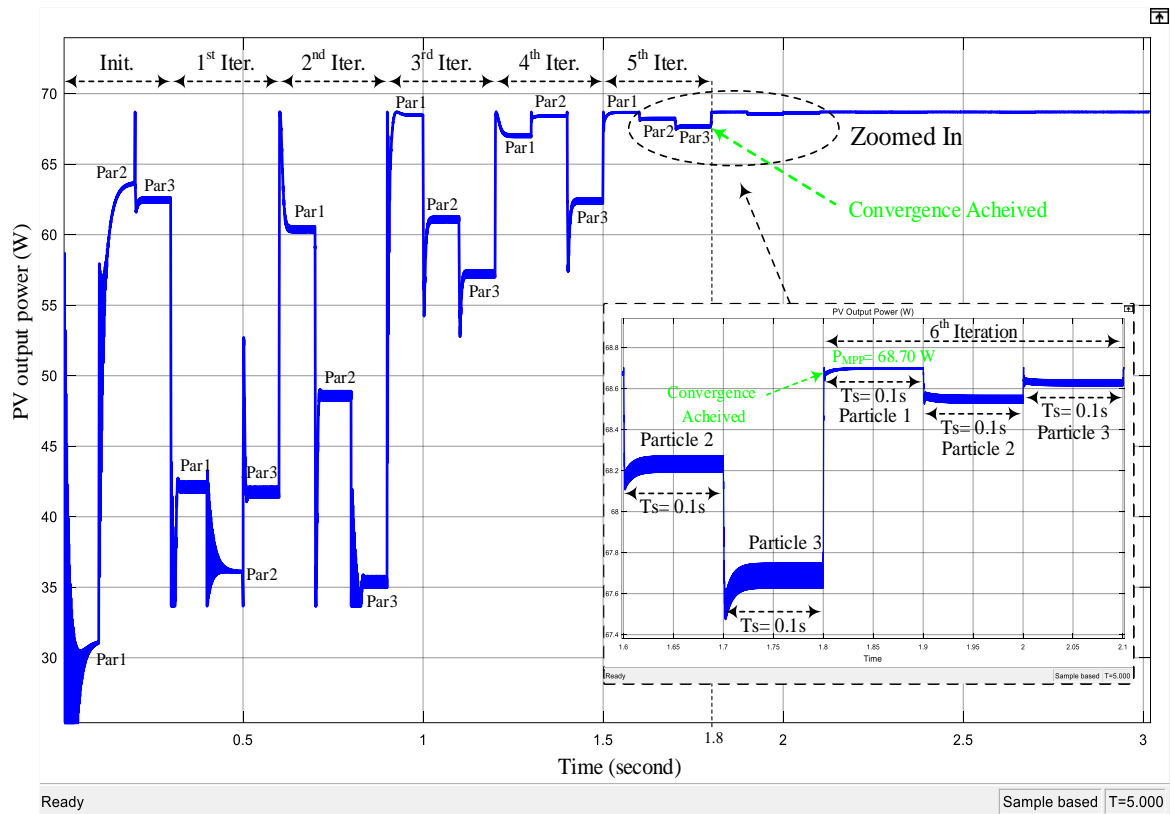


Figure 7. An example of a successful GP tracking using the PSO based MPPT

3. RESULTS AND DISCUSSION

The simulations were carried out using a computer equipped with 64-bit OS Windows 10 Professional with Intel(R) Core (TM) i5-3470 CPU @ 3.20 GHz processor and 8 GB of physical memory. Meanwhile, the PSO algorithm was coded in m-file of MATLAB environment and was tested on three PSC problems (pattern 1, 2 and 3). First, to study the impact of various NP values on successful PSO in solving the PSC problems, NP was varied in the sequence of 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 35, 40, 45 and 50. Other PSO parameters were as discussed in Section 2.2. Due to randomness in the initialization stage as reflected in (3), the distribution of particles in the SS was different at each simulation trial. With a lower NP, the probability for the particles to be initialized from a bad location is high, where bad location can refer to an area which is far from the GP location, or the area that only contain a GP for a certain case only. Particle initialization from a bad location may increase the probability of the PSO getting trapped at a LP. For example, when the NP is equal to 3, the distribution of the particles in the SS generated by the first 100 trials of (3) is as shown in Figure 8. For analysis purpose, the SS was divided into three sections (1st, 2nd and 3rd section). A bad initial location of particles was noted at the 65th trial, where the particles generated were too close to each other around the center of the 3rd section of the SS, *i.e.* $X_{\text{Initial_bad}} = [89.26, 92.36, 96.77]$. Therefore, the exploration process during optimization should focus more on the 3rd section of the SS, which only contains a GP for pattern 3.

To prove this concept, the PSO was initialized with $X_{\text{Initial_bad}}$ point and was assigned to solve the three PSC problems. About 10,000 independent simulation trials of the PSO were carried out to obtain sufficient representative sample and the results are presented in Table 3. As expected, the PSO achieved 100% of SR in solving pattern 3 as the exploration process was initialized around its GP. On the contrary, the

probability of the PSO to identify the GP for pattern 1 and pattern 2 was very low, which were 0.03% and 4.97%, respectively.

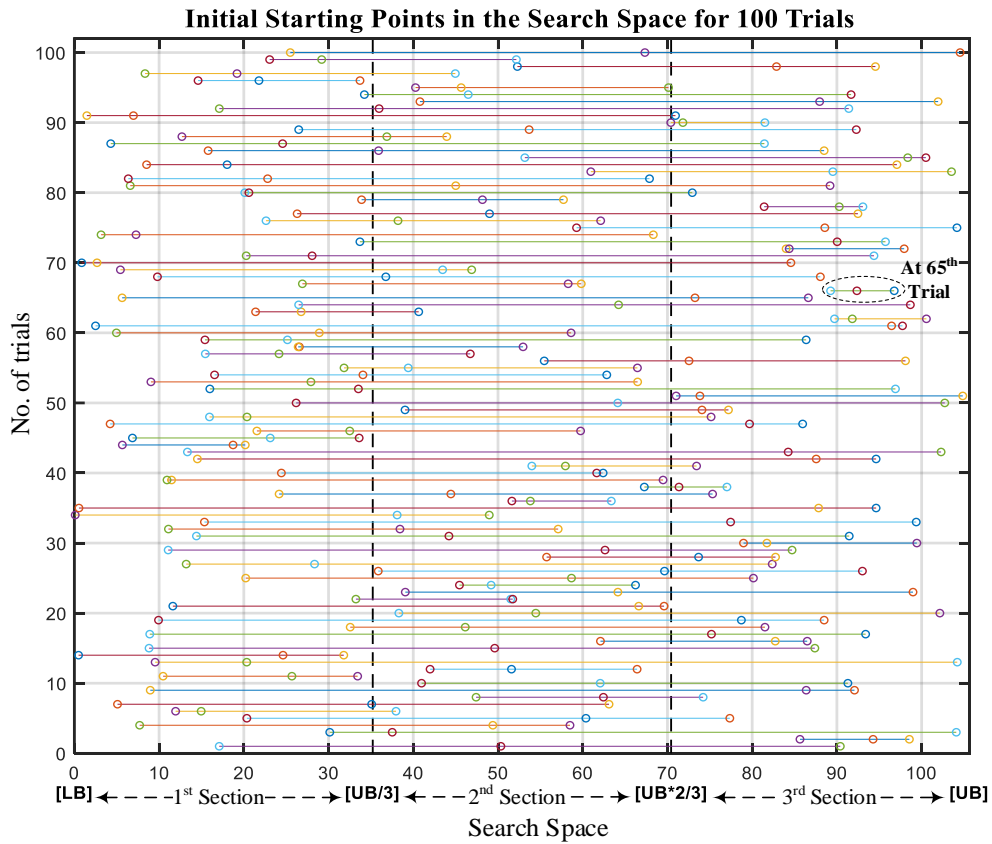


Figure 8. The plot of the particle distribution in the SS for the first 100 trials of (3)

Table 3. SR of the PSO when equipped with bad initial points of $X_{Initial\ bad} = [89.26, 92.36, 96.77]$

Initialization	PSCs	Global peak (GP) V_{MPP}	P_{MPP}	Successful runs of PSO in finding GP (% out of 10,000 trials)
$X_{Initial\ bad} = [89.26, 92.36, 96.77]$	Pattern 1 (Left GP)	12.53 V	68.70 W	3 (0.03%)
	Pattern 2 (Center GP)	52.23 V	169.90 W	497 (4.97%)
	Pattern 3 (Right GP)	93.47 V	238.58 W	10,000 (100%)

As recognized, a large NP is vital to reduce the probability of bad initial points or to distribute the particles more evenly in the SS. To prove this concept, 10 million independent trials of (3) were performed for each different NP as a large number of trials can produce stable probability distribution (PDIST). Subsequently, the PDIST of particles was analyzed and grouped according to the three sections of the SS. The overall results are as shown in Table 4.

As can be observed in Table 4, when the NP is equal to 3, the percentage of particles distributed only in each section was about 3.7% (out of 10 million independent trials). Meanwhile, the percentage of particles scattered in all three sections of the SS, $PDIST_{All}$ (at least one particle at each section) was about 44.4%. As the NP increases up to 30, the $PDIST_{All}$ percentage increases logarithmically as exhibited in Plot 1 (dotted green line) of Figure 9 until an optimum level of 100% was reached, where each section was occupied with at least one particle. To investigate the impact of increasing NP on the SR, the PSO was equipped with different NP in solving the three PSC problems. Again, 10,000 independent trials of the PSO were performed with different NP values. The number of successful trials, *i.e.* when the PSO solves with a required accuracy of 0.1 W of the true GP were recorded as shown in Table 5.

Table 4. PDIST of particles in the SS with different NP of (3)

SS Section	NP															
	3	4	5	6	7	8	9	10	15	20	25	30	35	40	45	50
1 st section [0, UB/3]	3.70	1.24	0.41	0.14	0.05	0.01	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
2 nd section [UB/3, 2UB/3]	3.72	1.23	0.41	0.14	0.05	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
3 rd section [2UB/3, UB]	3.71	1.24	0.41	0.14	0.05	0.02	0.01	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
All sections, PDIST _{All} [0, UB]	44.44	61.69	74.05	82.58	88.33	92.20	94.80	96.53	99.55	99.94	99.99	100	100	100	100	100

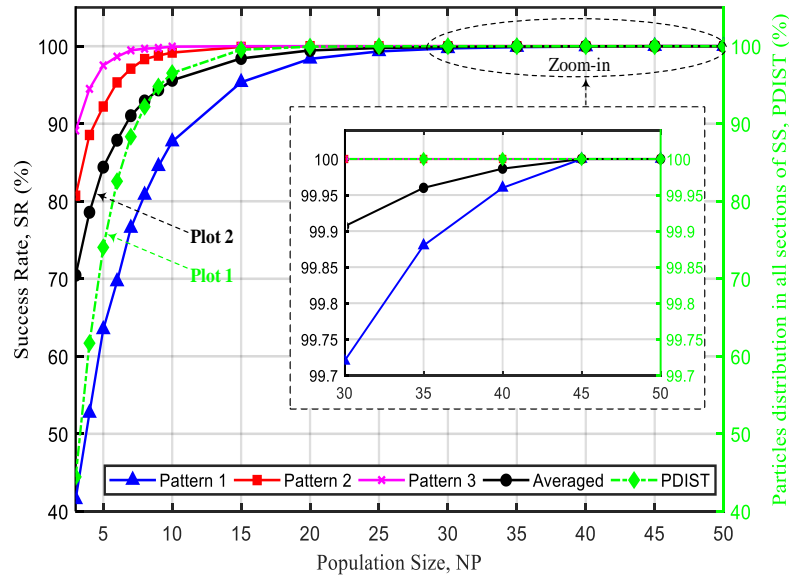


Figure 9. Probability plot of particle distribution for all SS sections and the trend of the PSO success rate

Table 5. SR of the PSO for different NP and PSCs

PSCs	NP															
	3	4	5	6	7	8	9	10	15	20	25	30	35	40	45	50
Pattern 1 (Left GP)	4155	5273	6348	6964	7659	8084	8452	8769	9537	9838	9933	9972	9988	9996	10000	10000
(SR % out of 10,000 trials)	(41.55)	(52.73)	(63.48)	(69.64)	(76.59)	(80.84)	(84.52)	(87.69)	(95.37)	(98.38)	(99.33)	(99.72)	(99.88)	(99.96)	(100)	(100)
Pattern 2 (Center GP)	8073	8856	9223	9534	9710	9834	9879	9918	9991	10000	10000	10000	10000	10000	10000	10000
(SR % out of 10,000 trials)	(80.73)	(88.56)	(92.23)	(95.34)	(97.10)	(98.34)	(98.79)	(99.18)	(99.91)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
Pattern 3 (Right GP)	8912	9451	9752	9865	9947	9969	9978	9994	10000	10000	10000	10000	10000	10000	10000	10000
(SR % out of 10,000 trials)	(89.12)	(94.51)	(97.52)	(98.65)	(99.47)	(99.69)	(99.78)	(99.94)	(100)	(100)	(100)	(100)	(100)	(100)	(100)	(100)
Average, SR _{Ave}	21140	23580	25323	26363	27316	27887	28309	28681	29528	29838	29933	29972	29988	29996	30000	30000
(SR % out of 10,000 trials)	(70.47)	(78.60)	(84.41)	(87.88)	(91.05)	(92.96)	(94.36)	(95.60)	(98.43)	(99.46)	(99.78)	(99.91)	(99.96)	(99.99)	(100)	(100)

As can be observed in Table 5, when the NP is equal to 3, in solving pattern 1, the number of PSO successful runs was 4,155 out of 10,000 trials which was equal to 41.55% of SR. In contrast, the SR in solving pattern 2 and pattern 3 was 80.73% (8,073 trials out of 10,000 trials) and 89.12% (8,912 trials out of 10,000 trials), respectively. The results implied that among the three PSC problems, pattern 1 was the most difficult one to solve. On average, the SR of PSO in solving all three problems when the NP is equal to 3 was 70.47% (21,140 trials out of 30,000 trials). The mean SR of the PSO increased logarithmically with the NP and reached a maximum of 100% at the NP equal to 45 as demonstrated in Plot 2 (black line) of Figure 9. It can be observed that the trend of Plot 1 and 2 is consistent which implied that a greater distribution of particles in all sections of SS leads to higher detection of GP. Therefore, a larger NP is indispensable to achieve a higher SR. Also, it can be concluded that to ensure 100% of SR in GP identification for various PSC problems, the NP should be specified at 45. This finding is in agreement with that reported in a previous study where the NP was proposed to be in the range from 20 to 50 particles. However, a larger NP can contribute to a higher CT. In the actual MPPT application, the detection of the GP should be quick since PV

systems are frequently subjected to fast-changing partial shading conditions. Hence, a trade-off has to be performed between the SR and CT in determining the optimal number of NP where in this study, an empirical trade-off based on the PI as in (7) was conducted. First, the mean number of iterations that reached the near GP solution first was obtained from the total successful runs of the PSO. A convergence plot of best fitness until the maximum iteration number of 50 (stopping criterion) for the NP of 3 in solving pattern 1 is shown in Figure 10.

As demonstrated in Figure 10, the dotted horizontal green line represents the true GP value of 68.70 W; the blue lines are the convergence plot of the best fitness for every 4,155 successful runs as shown in Table 5, and the black line is the mean convergence plot values. As can be observed in zoom-in view of Figure 10, the PSO reached the near true GP solution (*i.e.* when $GP - P_{PVmax} < 0.1W$) after an average convergence iteration, $ITER_{Con}$ of 26. The complete results of $ITER_{Con}$ for other NP and PSC problems are tabulated in Table 6 and plotted in Figure 11. The $ITER_{Con}$ values were noted to decrease as the NP increases for all the PSC problems solved. Next, the CT of the PSO was calculated using (6) as formulated in section 2.3. The overall CT results for different NP values and PSC problems (pattern 1, 2 and 3) are tabulated in Table 7, and the plot of CT versus varied NP is depicted in Figure 12. As can be seen for all three problems, the CT showed an increasing trend as the NP increased. The following example demonstrates how to calculate the CT for pattern 1 using the NP of 3 and the $ITER_{Con}$ of 26 from Table 6.

$$CT_{Ave}^1 = ITER_{Con}^1 \times NP \times T_{S_MPPT} = (26) \times (3) \times (0.1) = 7.8 \text{ s}$$

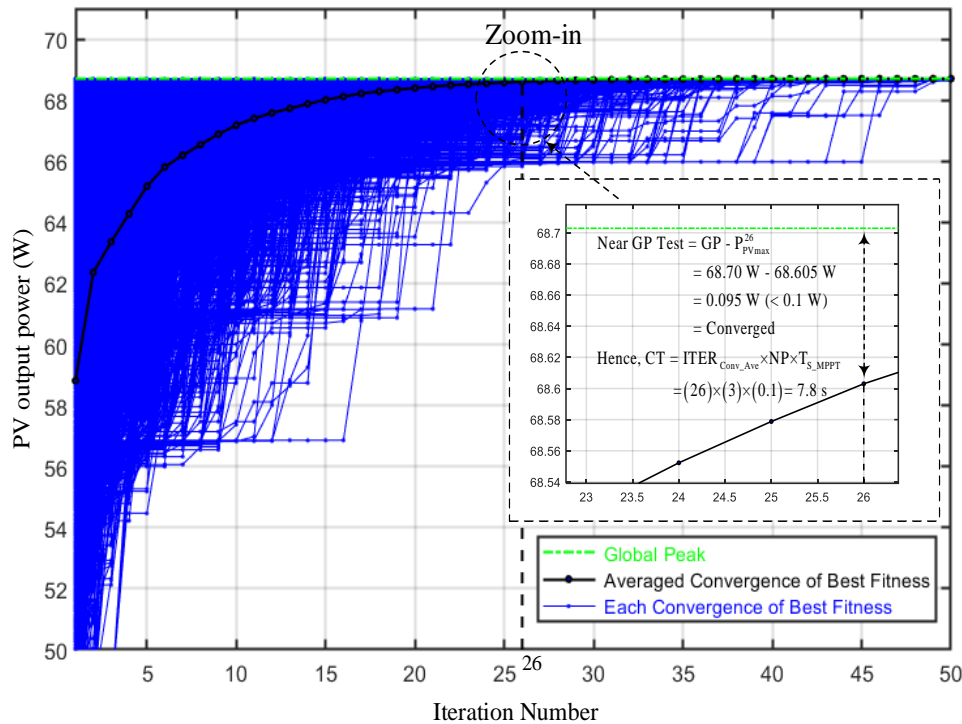


Figure 10. Convergence plot of best fitness for pattern 1 when the PSO was equipped with NP of 3

Table 6. $ITER_{Con}$ of the PSO for different NP and PSCs

PSCs	NP															
	3	4	5	6	7	8	9	10	15	20	25	30	35	40	45	50
Pattern 1 (Left GP)	26	25	24	23	23	22	21	20	16	14	12	10	9	8	8	7
Pattern 2 (Center GP)	22	21	19	17	16	15	14	13	10	8	6	6	5	5	4	4
Pattern 3 (Right GP)	20	18	16	14	13	12	11	10	8	6	5	5	4	4	3	3
Average, $ITER_{Con_Ave}$	22.7	21.3	19.7	18.0	17.3	16.3	15.3	14.3	11.3	9.3	7.7	7.0	6.0	5.7	5.0	4.7

Table 7. CT of the PSO for different NP and PSCs

PSCs	NP															
	3	4	5	6	7	8	9	10	15	20	25	30	35	40	45	50
Pattern 1 (Left GP)	7.8	10.0	12.0	13.8	16.1	17.6	18.9	20.0	24.0	28.0	30.0	30.0	31.5	32.0	36.0	35.0
Pattern 2 (Center GP)	6.6	8.4	9.5	10.2	11.2	12.0	12.6	13.0	15.0	16.0	15.0	18.0	17.5	20.0	18.0	20.0
Pattern 3 (Right GP)	6.0	7.2	8.0	8.4	9.1	9.6	9.9	10.0	12.0	12.0	12.5	15.0	14.0	16.0	13.5	15.0
Average, CT _{Ave}	6.8	8.5	9.8	10.8	12.1	13.1	13.8	14.3	17.0	18.7	19.2	21.0	21.0	22.7	22.5	23.3

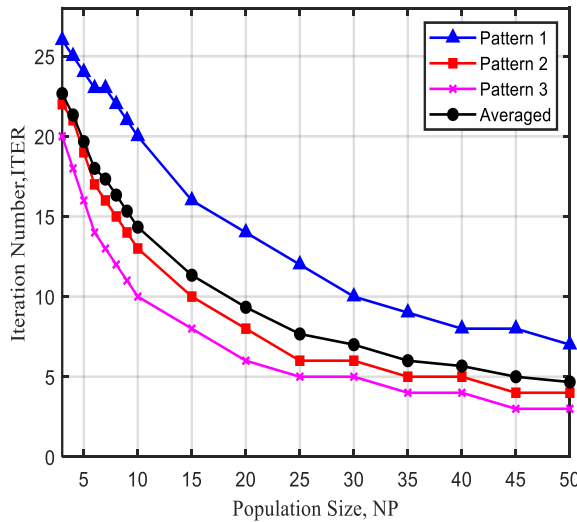


Figure 11. Plot of ITERCon versus NP

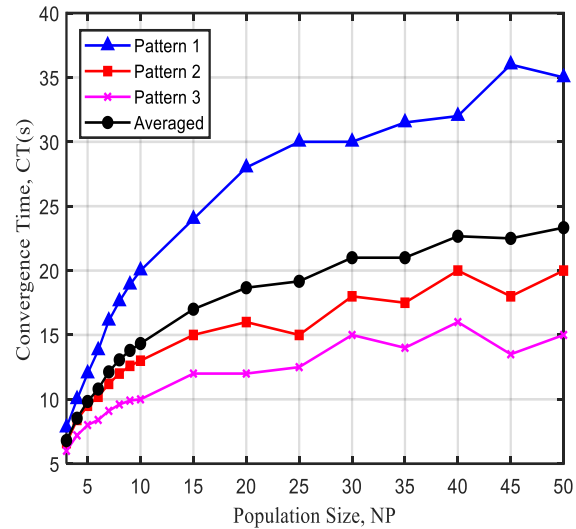


Figure 12. Plot of CT versus NP

Finally, the PI as derived from (7) was used to determine the optimal number of NP. An example of PI calculation for solving pattern 1 with the NP equals to 3, SR equals to 4,155 in Table 5, and CT equals to 7.8 s in Table 7 is:

$$PI^1 = k_1 \times (SR^1/N_7^1) + k_2 \times (CT_{NP=3}^1/CT_{Min}^1) = 0.9 \times (4,155/10,000) + 0.1 \times (7.8/7.8) = 0.4740$$

The complete results for other NP values and PSC patterns are tabulated in Table 8 and bold values indicate the maximum value of PI for each tested PSC problem. The optimum NP for PSO in solving Pattern 1 was noted to be 40 which produced the highest PI value of 0.9240 compared to other NP values. Meanwhile, for Pattern 2 and 3, the optimum NP was 25 and 7, respectively. As no prior information was available, the mean PI values for each NP for all three PSC problems were determined and presented in Table 8. Consequently, an optimum NP of 25 (PI=0.9373) was found for the PSO to successfully solve all the PSC problems. The relationship between the PI and NP is portrayed in Figure 13 where PI increases with increasing NP for all three problems solved. Furthermore, as can be seen from the zoom-in image of Figure 13, NP greater than the optimum value of 25 resulted in no significant improvement in PI.

For a clear observation of NP impacts on the PSO, PDIST_{All} in Table 4, SR_{Ave} in Table 5, CT_{Ave} in Table 6, ITER_{Con_Ave} in Table 7, and PI_{Ave} in Table 8 were presented in Figure 14. Both PDIST_{All} (dotted green line) and SR_{Ave} (blue line) were found to increase logarithmically with increasing NP. The trend implied that higher NP distributes particles more evenly over the SS which in turn increase the SR. Furthermore, as NP increased from 3 to 50, the ITER_{Con_Ave} decreased exponentially from 22.7 to 4.7, while CT_{Ave} increased logarithmically from 6.8 s to 23.3 s. The finding suggested that higher NP improves the iteration convergence speed of the PSO in identifying the GP. However, this inevitably led to a longer CT due to the use of a higher number of NP. Hence, an empirical trade-off method based on the PI with the incorporation of SR and CT was employed in the present study. From the zoom-in image of Figure 14, the highest average PI of 0.9373 was achieved when NP was 25, the optimal value.

Table 8. PI of the PSO for different NP and PSCs

PSCs	NP															
	3	4	5	6	7	8	9	10	15	20	25	30	35	40	45	50
Pattern 1 (Left GP)	0.4740	0.5526	0.6363	0.6833	0.7378	0.7719	0.8019	0.8282	0.8908	0.9133	0.9200	0.9235	0.9237	0.9240	0.9217	0.9223
Pattern 2 (Center GP)	0.8266	0.8756	0.8995	0.9228	0.9328	0.9401	0.9415	0.9434	0.9432	0.9413	0.9440	0.9367	0.9377	0.9330	0.9367	0.9330
Pattern 3 (Right GP)	0.9021	0.9339	0.9527	0.9593	0.9612	0.9597	0.9586	0.9595	0.9500	0.9500	0.9480	0.9400	0.9429	0.9375	0.9444	0.9400
Average, PI_{Ave}	0.7342	0.7874	0.8295	0.8551	0.8772	0.8905	0.9007	0.9104	0.9280	0.9348	0.9373	0.9334	0.9348	0.9315	0.9343	0.9318

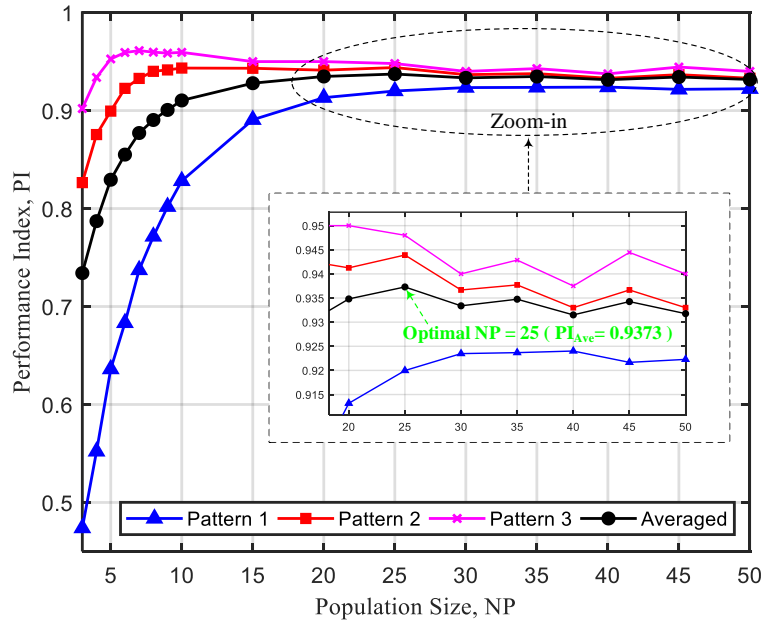


Figure 13. Plot of PI versus NP

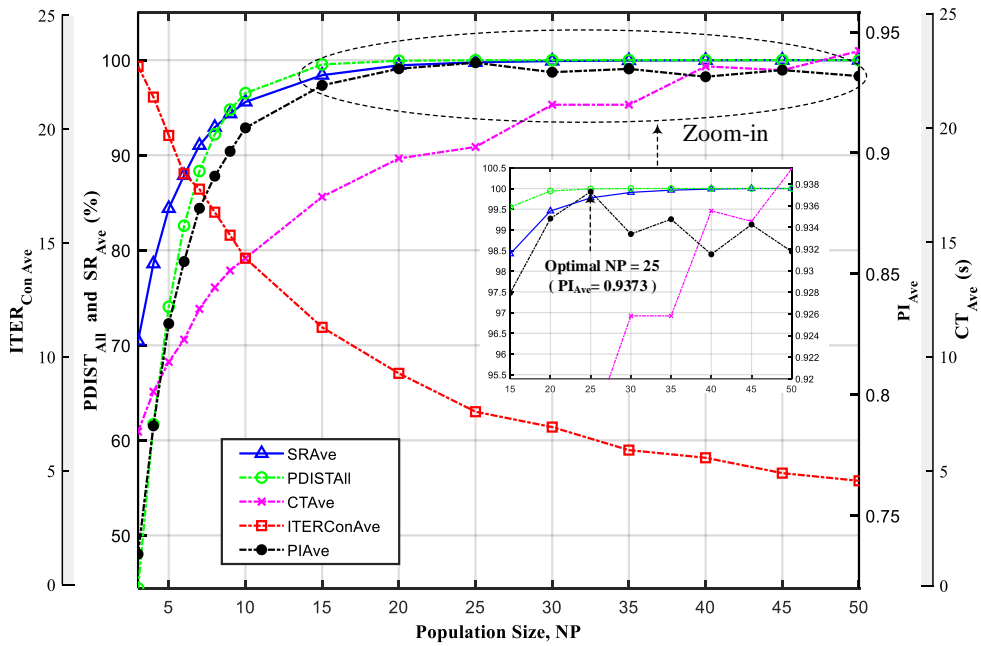


Figure 14. Plot of $PDIST_{All}$, SR_{Ave} , CT_{Ave} , $ITER_{Con_Ave}$ and PI_{Ave} versus NP

4. CONCLUSION

The impact of NP on the successful application of PSO algorithm in solving various PSCs has been verified in the present study. The PSO was evaluated with 16 different NPs of 3, 4, 5, 6, 7, 8, 9, 10, 15, 20, 25, 30, 35, 40, 45 and 50 for three PSC problems. All the PSC problems represent the case of no prior information about the location of GP in the SS. The PSC problems were categorized into pattern 1, 2 and 3 with the GP located at the left side (1st section), middle (2nd section) and the right side (3rd section) of the SS, respectively. To reduce the statistical errors, 10,000 independent runs were performed for each NP. Based on the empirical results, NP was found to increase resulting in an increased success rate of the PSO. The results further confirmed previous findings where large NP was found to increase the probability of successful optimization. As expected, the convergence time also increased as a function of NP which caused the implementation of fast MPPT impractical. Therefore, an empirical trade-off methodology based on PI indicator was proposed to evaluate and select the optimum number of NP for the PSO considering the weighted importance of both success rate and convergence time. It was found that to achieve the desired performance of PSO in solving pattern 1, the optimal NP should be 40 which can produce the highest PI value of 0.9240. Meanwhile, for pattern 2 and 3, the optimal NP should be 25 and 7, respectively. It can be concluded that the selection of optimal NP should be specific for different cases or problems, with pattern 1 being the most difficult one. Overall, all three PSC problems can be solved successfully if the optimal NP is specified at 25 to produce the best average PI value of 0.9373. The optimal NP of 25 is consistent with the range of NP (20-50 particles) reported in the literature.

ACKNOWLEDGEMENTS

This research was supported and funded by the Research Management Centre (RMC), Universiti Teknologi MARA (UiTM), Shah Alam, Malaysia under the MyRA research grant scheme (Project Number: 600-RMC/MYRA 5/3/LESTARI (015/2020)).





REFERENCES

- [1] M. Louzazni, E. H. Aroudam, and H. Yatimi, "Modeling and simulation of a solar power source for a clean energy without pollution," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 3, no. 4, Jul. 2013, doi: 10.11591/ijece.v3i5.3639.
- [2] H. R. Mohammadi and A. Akhavan, "A new control method for grid-connected PV system based on quasi-Z-source cascaded multilevel inverter using evolutionary algorithm," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 6, no. 1, pp. 109–120, Mar. 2015, doi: 10.11591/ijpeds.v6.i1.pp109-120.
- [3] N. Hashim, Z. Salam, and N. F. N. Ismail, "An improved evolutionary programming (IEP) method under the EN 50530 dynamic MPPT efficiency Test," in *2019 IEEE Conference on Energy Conversion (CENCON)*, Oct. 2019, pp. 147–152, doi: 10.1109/CENCON47160.2019.8974841.
- [4] Z. Salam, Z. Ramli, J. Ahmed, and M. Amjad, "Partial shading in building integrated PV system: Causes, effects and mitigating techniques," *International Journal of Power Electronics and Drive Systems (IJPEDS)*, vol. 6, no. 4, pp. 712–722, Dec. 2015, doi: 10.11591/ijpeds.v6.i4.pp712-722.
- [5] K. Ishaque, Z. Salam, A. Shamsudin, and M. Amjad, "A direct control based maximum power point tracking method for photovoltaic system under partial shading conditions using particle swarm optimization algorithm," *Applied Energy*, vol. 99, pp. 414–422, Nov. 2012, doi: 10.1016/j.apenergy.2012.05.026.
- [6] D. Yousri, D. Allam, and M. B. Eteiba, "Optimal photovoltaic array reconfiguration for alleviating the partial shading influence based on a modified harris hawks optimizer," *Energy Conversion and Management*, vol. 206, Feb. 2020, doi: 10.1016/j.enconman.2020.112470.
- [7] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," in *Advances in Fuzzy Systems Applications and Theory*, 1996, pp. 775–782.
- [8] G. Dileep and S. N. Singh, "Application of soft computing techniques for maximum power point tracking of SPV system," *Solar Energy*, vol. 141, pp. 182–202, Jan. 2017, doi: 10.1016/j.solener.2016.11.034.
- [9] J. Ahmed and Z. Salam, "A critical evaluation on maximum power point tracking methods for partial shading in PV systems," *Renewable and Sustainable Energy Reviews*, vol. 47, pp. 933–953, Jul. 2015, doi: 10.1016/j.rser.2015.03.080.
- [10] J. P. Ram, T. S. Babu, and N. Rajasekar, "A comprehensive review on solar PV maximum power point tracking techniques," *Renewable and Sustainable Energy Reviews*, vol. 67, pp. 826–847, Jan. 2017, doi: 10.1016/j.rser.2016.09.076.
- [11] 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 (IJPEDS)*, vol. 10, no. 1, pp. 548–561, Mar. 2019, doi: 10.11591/ijpeds.v10.i1.pp548-561.
- [12] S. D. Al-Majidi, M. F. Abbod, and H. S. Al-Raweshidy, "A particle swarm optimisation-trained feedforward neural network for predicting the maximum power point of a photovoltaic array," *Engineering Applications of Artificial Intelligence*, vol. 92, Jun. 2020, doi: 10.1016/j.engappai.2020.103688.
- [13] M. Kermadi, Z. Salam, J. Ahmed, and E. M. Berkouk, "A high-performance global maximum power point tracker of PV system for rapidly changing partial shading conditions," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2236–2245, Mar. 2021, doi: 10.1109/TIE.2020.2972456.
- [14] M. Ren, X. Huang, X. Zhu, and L. Shao, "Optimized PSO algorithm based on the simplicial algorithm of fixed point theory," *Applied Intelligence*, vol. 50, no. 7, pp. 2009–2024, Jul. 2020, doi: 10.1007/s10489-020-01630-6.
- [15] S. K. Gopalakrishnan, S. Kinattungal, S. P. Simon, and K. Ark Kumar, "Enhanced energy harvesting from shaded PV systems using an improved particle swarm optimisation," *IET Renewable Power Generation*, vol. 14, no. 9, pp. 1471–1480, Jul. 2020, doi: 10.1049/iet-rpg.2019.0936.




- [16] N. Hashim, Z. Salam, N. F. Nik Ismail, and D. Johari, "New deterministic initialization method for soft computing global optimization algorithms," *Indonesian Journal of Electrical Engineering and Computer Science (IJECS)*, vol. 18, no. 3, Jun. 2020, doi: 10.11591/ijeecs.v18.i3.pp1607-1615.
- [17] Z. Li-ping, Y. Huan-jun, and H. Shang-xu, "Optimal choice of parameters for particle swarm optimization," *Journal of Zhejiang University-SCIENCE A*, vol. 6, no. 6, pp. 528–534, Jun. 2005, doi: 10.1631/jzus.2005.A0528.
- [18] J. C. Bansal, *Evolutionary and swarm intelligence algorithms*, vol. 779. Cham: Springer International Publishing, 2019.
- [19] M. Saravanan, S. M. R. Slochanal, P. Venkatesh, and J. P. S. Abraham, "Application of particle swarm optimization technique for optimal location of FACTS devices considering cost of installation and system loadability," *Electric Power Systems Research*, vol. 77, no. 3–4, pp. 276–283, Mar. 2007, doi: 10.1016/j.epr.2006.03.006.
- [20] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence*, 1998, pp. 69–73, doi: 10.1109/ICEC.1998.699146.
- [21] Y. E. Yildiz and A. O. Topal, "Large scale continuous global optimization based on micro differential evolution with local directional search," *Information Sciences*, vol. 477, pp. 533–544, Mar. 2019, doi: 10.1016/j.ins.2018.10.046.
- [22] V. Stanovov, S. Akhmedova, and E. Semenkin, "Selective pressure strategy in differential evolution: exploitation improvement in solving global optimization problems," *Swarm and Evolutionary Computation*, vol. 50, Nov. 2019, doi: 10.1016/j.swevo.2018.10.014.
- [23] A. Kumar, R. K. Misra, D. Singh, S. Mishra, and S. Das, "The spherical search algorithm for bound-constrained global optimization problems," *Applied Soft Computing*, vol. 85, Dec. 2019, doi: 10.1016/j.asoc.2019.105734.
- [24] H. Li, D. Yang, W. Su, J. Lu, and X. Yu, "An overall distribution particle swarm optimization MPPT algorithm for photovoltaic system under partial shading," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 1, pp. 265–275, Jan. 2019, doi: 10.1109/TIE.2018.2829668.
- [25] Y. W. Guo, W. D. Li, A. R. Mileham, and G. W. Owen, "Applications of particle swarm optimisation in integrated process planning and scheduling," *Robotics and Computer-Integrated Manufacturing*, vol. 25, no. 2, pp. 280–288, Apr. 2009, doi: 10.1016/j.rcim.2007.12.002.
- [26] K. Ishaque, Z. Salam, and Syafaruddin, "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, Sep. 2011, doi: 10.1016/j.solener.2011.06.008.
- [27] M. AlShabi, C. Ghenai, M. Bettayeb, and F. Faraz Ahmad, "Estimating one-diode-PV model using autonomous groups particle swarm optimization," *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 10, no. 1, pp. 166–174, Mar. 2021, doi: 10.11591/ijai.v10.i1.pp166-174.
- [28] A. H. Ali, H. S. Hamad, and A. A. Abdnlrazzaq, "Performance investigation of grid connected photovoltaic system modelling based on MATLAB simulation," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, no. 6, pp. 4847–4854, Dec. 2018, doi: 10.11591/ijece.v8i6.pp4847-4854.
- [29] A. P. Yoganandini and G. S. Anitha, "A modified particle swarm optimization algorithm to enhance MPPT in the PV array," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 10, no. 5, pp. 5001–5008, Oct. 2020, doi: 10.11591/ijece.v10i5.pp5001-5008.
- [30] "Solarex MSX-60 w/ junction box," *Solarelectricsupply.com*. <https://www.solarelectricsupply.com/solar-panels/solarex/solarex-msx-60-w-junction-box> (accessed Apr. 21, 2022).
- [31] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 1995, pp. 39–43, doi: 10.1109/MHS.1995.494215.
- [32] A. R. Bhatti *et al.*, "Optimized sizing of photovoltaic grid-connected electric vehicle charging system using particle swarm optimization," *International Journal of Energy Research*, vol. 43, no. 1, pp. 500–522, Jan. 2019, doi: 10.1002/er.4287.
- [33] A. Nickabadi, M. M. Ebadzadeh, and R. Safabakhsh, "A novel particle swarm optimization algorithm with adaptive inertia weight," *Applied Soft Computing*, vol. 11, no. 4, pp. 3658–3670, Jun. 2011, doi: 10.1016/j.asoc.2011.01.037.
- [34] A. P. Piotrowski, J. J. Napiorkowski, and A. E. Piotrowska, "Population size in particle swarm optimization," *Swarm and Evolutionary Computation*, vol. 58, Nov. 2020, doi: 10.1016/j.swevo.2020.100718.
- [35] V. Garg and K. Deep, "Performance of laplacian biogeography-Based optimization algorithm on CEC 2014 continuous optimization benchmarks and camera calibration problem," *Swarm and Evolutionary Computation*, vol. 27, pp. 132–144, Apr. 2016, doi: 10.1016/j.swevo.2015.10.006.
- [36] K. Deep and M. Thakur, "A new crossover operator for real coded genetic algorithms," *Applied Mathematics and Computation*, vol. 188, no. 1, pp. 895–911, May 2007, doi: 10.1016/j.amc.2006.10.047.
- [37] N. Hashim, Z. Salam, D. Johari, and N. F. Nik 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 (IJPEDS)*, vol. 9, no. 3, pp. 1038–1050, Sep. 2018, doi: 10.11591/ijpeds.v9.i3.pp1038-1050.

BIOGRAPHIES OF AUTHORS






Norazlan Hashim     received his Ph.D. degree in Electrical Engineering from Universiti Teknologi Malaysia (UTM), Skudai in 2022. He obtained his B.Eng. and M.Eng. degrees in Electrical Engineering from University of Malaya (UM), Kuala Lumpur, in 2001 and 2007, respectively. He is currently a Senior Lecturer in the School of Electrical Engineering at the Universiti Teknologi MARA (UiTM), Shah Alam, Malaysia. His research interests include maximum power point tracking techniques, power electronic converters, artificial intelligence algorithms and PV systems. He can be contacted at email: azlan4477@uitm.edu.my.






Nik Fasdi Nik Ismail    received the B.Eng. degree in communication and information engineering from Tokyo Denki University, Tokyo, Japan in 2002 and the M.S. and Ph.D. Degrees in electric power system and power electronics from University Malaya, Kuala Lumpur, Malaysia in 2010 and 2018, respectively. Currently, he is Senior Lecturer at the School of Electrical Engineering, UiTM, Malaysia. His research interests include power electronics, power drives and inverters, power system reliability and fuel cell system. He can be contacted at email: nikfasdi@uitm.edu.my.






Dalina Johari    received her Ph.D. degree in Engineering Science from Uppsala University, Sweden in 2017. She obtained her M.Sc and B.Eng. degrees in Electrical Engineering from Universiti Teknologi MARA (UiTM), Malaysia in 2008 and the University of Liverpool, UK in 1999, respectively. She currently works as a Senior Lecturer at UiTM, Shah Alam, Malaysia. She has experience working as an engineer in the telecommunication industry for about 6 years. Her research interests include high voltage, lightning physics, lightning protection, power system and artificial intelligence. She can be contacted at email: dalinaj@uitm.edu.my.



Ismail Musirin    holds a Ph.D. in Power System from Universiti Teknologi MARA (UiTM), Malaysia. He is currently a Professor of Power System at the School of Electrical Engineering (formerly known as the Faculty of Electrical Engineering), College of Engineering, UiTM and headed the Power System Operation Computational Intelligence (POSC) Research Group. He has published over 350 papers in international indexed journals and conferences. He has chaired more than 20 international conferences since 2007. To date, he has delivered keynote speeches at Cambridge University, United Kingdom, Dubai, Korea, India, and Malaysia. He has also been given the opportunity to evaluate research grants at the national and international levels. His research interest includes artificial intelligence, optimization techniques, power system analysis, renewable energy optimization, distributed generation, and power system stability. He is a professional engineer and a senior member of the International Association of Computer Science and Information Technology (IACSIT), the Artificial Immune System Society (ARTIST), and the International Association of Engineers (IAENG), Hong Kong. He can be contacted at email: ismailbm@uitm.edu.my or ismaibm1@gmail.com.



Azhan Ab. Rahman    graduated with a B.Eng. in Electrical and Electronics from Cardiff University in 2006, M.Eng. Studies in Electrical Engineering from University of Wollongong in 2008 and a Ph.D. in Electrical Engineering from Universiti Teknologi Malaysia (UTM), Skudai in 2021. A registered electrical energy manager (REEM) with the Energy Commission (EC) since 2016, his main research interests are in the areas of renewable energy, energy efficiency, energy management and power engineering. Presently, he is a lecturer at the Faculty of Electrical and Electronic Engineering Technology, UTeM. He can be contacted at his email: azhanrahman@utem.edu.my.