❏ 166

Estimating one-diode-PV model using autonomous groups particle swarm optimization

Mohammad AlShabi[1], Chaouki Ghenai[2], Maamar Bettayeb[3], Fahad Faraz Ahmad[4]
[1]Mechanical and Nuclear Engineering Department, University of Sharjah, Sharjah, United Arab Emirates
[2]Sustainable and Renewable Energy Engineering Department, University of Sharjah, Sharjah, United Arab Emirates
[2,4]Sustainable Energy Development Research Group, Research Institute for Sciences and Engineering, University of Sharjah, Sharjah, United Arab Emirates
[3]Electrical Engineering Department, University of Sharjah, Sharjah, United Arab Emirates
[3]Centre of Excellence in Intelligent Engineering systems (CEIES), King Abdulaziz University, Jeddah, Saudi Arabia

Article Info	ABSTRACT
	In this paper, the one-diode model of a photovoltaic PV solar cell (PVSC) is estimated for an experimental characteristic curves data by using a recently proposed version of the particle swarm optimization (PSO) algorithm, which is known as the autonomous groups particles swarm optimization (PSOAG). This meta-heuristic algorithm is used to identify the model of the PVSC. The PSOAG divides the particles into groups and then, uses different functions to tune the social and cognitive parameters of these groups. By using these groups, the performance of the PSO is improved in terms of convergence rate and escaping the local minima/maxima. Six versions of PSOAG algorithms were developed in this work. Therefore, nine versions of PSOAG, including these six algorithms and three newly developed PSOAG reported previously, will be used in this research to cover more social's behaviors. The results are compared to the original PSO and other versions of PSO like conventional and asymmetric time-varying accelerated coefficient PSOs, and the improved PSO. The result shows that the proposed methods improve the performance by up to 14% in terms of root mean squared error and maximum absolute error, and by up to 20% in term of convergence rate, when these were compared to the best results obtained from the other algorithms. *This is an open access article under the CC BY-SA license.*

Corresponding Author:

Mohammad AlShabi
Mechanical and Nuclear Engineering Department
University of Sharjah
Sharjah, United Arab Emirates
Email: malshabi@Sharjah.ac.ae

1. INTRODUCTION

Despite of uncertainty in environmental conditions, solar energy is getting more and more attention from researchers due to its characteristics of being economical, low maintenance cost and long-life span. The availability of solar irradiance and direct conversion of solar energy into electricity are also the other factors, which make photovoltaic systems more desirable than other alternative sources [1]. However, the efficiency of PV systems is directly affected by physical changes in the environment such as solar irradiance and temperature [2]. For reliable design and exploitation of maximum energy and energy security of PV systems, it is critical to devise a comprehensive and accurate mathematical model to predict the behavior of PV solar cell (PVSC) module under all environmental conditions [3].

Several mathematical models based on electrical equivalent circuits are reported in the literature such as one-diode model [4-5], two-diode model [6], multi-diode model [7], R_s-Model, and R_p model [4-7]. The one-diode model is most widely used and cited as it provides good trade-off between accuracy and complexity [8]. The one-diode model is governed by five parameters stated as reverse saturation (I_d), and photo-currents (I_{ph}), diode ideality factor (n), and series (R_s) and shunt (R_{sh}) resistances. The two-diode model provides more accuracy but also increases the complexity of model and is governed by seven parameters [6].

The estimation of characteristic parameters concise with real data is the main concern in mathematical modeling of PVSC. Unknown parameters can be extracted mainly by two methodologies classified as analytical and numerical approaches. In the analytical approach, the information provided by the manufacturer or I-V curve data is used to determine the unknown parameters [9]. The manufacturer usually provides the data only at standard testing conditions (STC) which made this approach inappropriate to be used for changing irradiance and temperature conditions. In addition, the simplifying assumptions considered during the formulation of analytical models lead to compromising the accuracy [10]. In order to overcome the disadvantages of analytical methods, several non-linear Curve fitting techniques are reported and categorized as numerical methods [11]. Numerical methods are further divided in conventional or iterative methods and metaheuristic methods. Newton Raphson [12] and Levenberg Marquardt algorithms are examples of conventional iterative methods [13-14]. In spite of the efficiency of conventional techniques, they converge prematurely around local minima and require continuity and differentiability of the functions. In addition, the efficiency of conventional iterative methods depends upon initial positions [15].

A wide variety of metaheuristic algorithms is presented in the literature to solve non-linearity optimization problems. These optimization techniques include genetic algorithm [15], particle swarm optimization (PSO) [16], hybrid particle swarm optimization [17], harmony search [18], differential evolution, adaptive differential evolution [19], cat swarm optimization algorithm [20], bacterial foraging optimization [21], biogeography-based optimization [22], flower pollination algorithm [23], artificial bee colony algorithm [24].

Metaheuristic algorithms are able to identify the parameters of a non-linear model in varying environmental conditions. PSO is a widely adopted optimization algorithm due to its simplicity and inexpensiveness in term of computational cost [25]. On the other hand, it expresses the disadvantage of converging to local minima/maxima and slow convergence rate (CR) resulting in low quality estimation of parameters [26]. To alleviate these issues, many improved versions have been presented including time varying accelerated coefficient PSO (PSOTAC), asymmetrical time varying accelerated coefficient particle swarm (PSOM) and autonomous group particle swarm optimization (PSOAG) [27-31]. In PSOTAC, the modification in positive acceleration constants are introduced during the optimization to overcome premature convergence [27]. The zero velocity particles are reinitializing with random velocity to overcome premature convergence [28]. In [29], the author adopted an adjustment strategy of asymmetrical time dependent acceleration coefficients to eliminate the local minima problem (PSOM). The improved version of PSOM is outlined in [30]. An autonomous group particle swarm optimization (PSOAG) has been developed considering the diversity in bird flocking or insect swarming [31]. The mathematical modeling of autonomous groups was presented, and results had shown good performance in term of avoiding local minima/maxima and convergence speed. Autonomous groups are employed in such a way to give independence to diverse particles in order to enhance the performance [31].

In this paper, for the first time, a recently developed autonomous group particle swarm optimization (PSOAG) [31] is employed to determine the five parameters of one diode model. The particles are divided into groups with tuned social and cognitive parameters. In this study, six new versions of PSOAG (PSOAG4 … PSOAG9) are developed along with three versions presented in [31] are studied. A brief comparison of the proposed methods with conventional PSO, PSOTAC [27-28], PSOM [29] and its improved version (PSOI) [30] is also included. The paper organization is: section 2 provides the research methodology, and the proposed algorithm is proposed in section 3. Results and discussion are presented in section 4. In the end, the study is concluded in section 5. The contributions of this work can be summarized as:
- Developing six new versions of the PSOAG to cover more social behaviours.
- Applying all versions of PSOAG; the new developed methods in this work, and the previously developed methods in [31], to find the parameters of a one diode PV model (ODM) from experimental data.
- Use statistical tools to examine the methods in terms of how many times the method reached the targets (TRT), max-min-mean of conducted iterations (MMM), root mean squared error (RMSE), CR, and maximum absolute error (MAE) in the results.

2. RESEARCH METHOD

Several representations were proposed to describe the PVSC. One of the widely used model is the SDM represented in Figure 1 and is given in [4-5]. The model represents the PVSC by (1).

$$I = I_{ph} - I_d \left(e^{\frac{q(V+IR_s)}{nkT}} - 1 \right) - \frac{V+IR_s}{R_{sh}} \qquad (1)$$
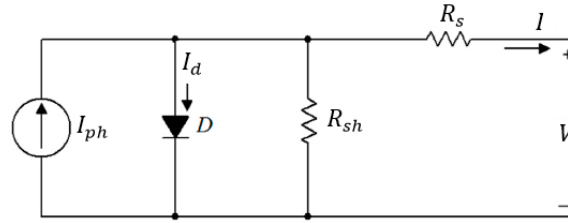

Figure 1. One-diode model [4-5]

Where q is the electron charge, k is the boltzmann's constant, T is the temperature in Kelvin, and V is the voltage of the PVSC in (V). The parameters: I_{ph}, I_d, R_s, R_{sh} and n are unknown and are needed to be obtained based on the curve of I-V characteristics. Once these parameters are identified, the entire cell performance could be predicted, and the maximum power could be achieved. Therefore, several works have been conducted to find these parameters as in [25-32]. By rearranging (1), the objective function can be obtained as [4-5]:

$$J = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(I_i - I_{ph} + I_d \left(e^{\frac{q(V_i+I_iR_s)}{nkT}} - 1 \right) + \frac{V_i+I_iR_s}{R_{sh}} \right)^2} \qquad (2)$$

Where N is the number of data available. The subscript i is a counter. The aim is to minimize J so that the exact five parameters are obtained when $J = 0$. Four well known versions of PSO algorithms; PSO, PSOAG1, …, PSOAG3, and six new algorithms; PSOAG4, …, PSOAG9, are used in this work to solve the PVSC problem.

2.1. Introduction to PSO

Particle swarm optimization is one of the most famous metaheuristic optimization techniques. It starts with choosing candidates to be a solution set for the problem, referred to as particles, and then improve these particles iteratively to enhance the performance. The particles move within a space trying to find the solution in their location; $pbest_i$, or the best location of the neighborhood with; $gbest_i$. Each particle memorizes its position, and its velocity, which are defined for n particles as $x_i = [x_{i1} \ x_{i2} \dots x_{in}]$, and $v_i = [v_{i1} \ v_{i2} \dots v_{in}]$, respectively. These are initialized randomly. Then, they are enhanced using [27-31]:

$$v_i^{t+1} = \omega v_i^t + c_1 \times rand \times (pbest_i - x_i^t) + c_2 \times rand \times (gbest_i - x_i^t) \qquad (3)$$

$$x_i^{t+1} = x_i^t + v_i^{t+1} \qquad (4)$$

Where ω is the inertial weight and it has a value of [0.4,0.9], c_1 and c_2 are the learning coefficient of the individual and group best solutions found, respectively, and $rand$ is a random number in [0,1]. These coefficients; ω, c_1 and c_2, are constant in the conventional PSO. However, it has been found that the performance becomes better when they are time varying [27-31], this is the essential motivation for developing the other versions discussed in this paper.

2.2. Proposed algorithm-autonomous groups PSO

In [31], the authors used the fact that the individuals are not quite similar in terms of ability or intelligence although they do their duties. Therefore, they divide the swarm into four groups, and each group has different learning coefficients c_1 and c_2; several functions are used such as linear, parabolic, quadratic, exponential, sinusoidal and logarithmic functions. These groups are called autonomous groups (AG), and the

method they represent is called PSOAG. Figure 2 shows the pseudo-code for PSOAG. Table 1 shows the coefficients for nine different versions of PSOAG referred to PSOAG1, PSOAG2, … PSOAG9, where the first three are derived in [31] and the rest are proposed for the first time in this paper.

```
Create n particles randomly.
While i ≤Max Iteration and cost function>0
        Calculate particles' fitness, pbest and gbest
        Match particles to their groups
        Update particles according to (3) and (4) and using Table 1
        Apply limits
        Evaluate fitness using (2)
        Update the solutions' lists by adding better positions and remove the worst positions (pbest, gbest).
Repeat
```

Figure 2. Pseudo-code for PSOAG

Table 1. Learning coefficients for the PSOAG1, PSOAG2 … PSOAG9

PSOAG#	Group	Learning Coefficient c_1	c_2
1 [31]	1	$-2.05S + 2.55$	$S + 1.25$
	2	$-2.05S + 2.55$	$2S^3 + 0.5$
	3	$-2S^3 + 2.5$	$S + 1.25$
	4	$-2S^3 + 2.5$	$2S^3 + 0.5$
2 [31]	1	$2.5 - S_1$	$0.5 + S_1$
	2	$-2S^3 + 2.5$	$2S^3 + 0.5$
	3	$0.5 + 2e^{-16S^2}$	$2.2 - 2e^{-16S^2}$
	4	$2.5 + 2S^2 - 4S$	$0.5 - 2S^2 + 4S$
3 [31]	1	$-2S^{1/3} + 1.95$	$2S^{1/3} + 0.05$
	2	$-2S^3 + 2.5$	$2S^3 + 0.5$
	3	$-2S^{1/3} + 1.95$	$2S^3 + 0.5$
	4	$-2S^3 + 2.5$	$2S^{1/3} + 0.05$
4	1	$-2.05S + 2.55$	$2.5 - 2S^2 + 4S$
	2	$-2.05S + 2.55$	$2.2 - 2e^{-16S^2}$
	3	$-2S^3 + 2.5$	$2S^3 + 0.5$
	4	$-2S^3 + 2.5$	$0.5 + S_1$
5	1	$-2.05S + 2.55$	$S + 1.25$
	2	$-2.05S + 2.55$	$2S^3 + 0.5$
	3	$-2S^3 + 2.5$	$S + 1.25$
	4	$-2S^3 + 2.5$	$2S^3 + 0.5$
6	1	$2.5 + 2S^2 - 4S$	$S + 1.25$
	2	$0.5 + 2e^{-16S^2}$	$2S^3 + 0.5$
	3	$-2S^3 + 2.5$	$S + 1.25$
	4	$2.5 - S_1$	$2S^3 + 0.5$
7	1	$2.5 - S_1$	$0.5 - 2S^2 + 4S$
	2	$2.5 - S_1$	$2.2 - 2e^{-16S^2}$
	3	$2.5 - S_1$	$2S^3 + 0.5$
	4	$2.5 - S_1$	$0.5 + S_1$
8	1	$-2S^{1/5} + 1.95$	$2S^5 + 0.5$
	2	$-2S^{1/5} + 1.95$	$2S^5 + 0.5$
	3	$-2S^5 + 2.5$	$2S^5 + 0.5$
	4	$-2S^5 + 2.5$	$2S^5 + 0.5$
9	1	F_1	F_2
	2	$-2S^{1/3} + 1.95$	$2S^4 + 0.5$
	3	$-2S^4 + 2.5$	$2S^4 + 0.5$
	4	$-2S^6 + 2.5$	$2S^6 + 0.5$

where $S = t/T$, t is the iteration number compared to the total iterations T

$$S_1 = \frac{2\log(t)}{\log(T)}, \ F_1 = 2.5 \cos\left(\frac{\pi S}{2}\right) e^{-16S^2}$$

$$F_2 = 0.5 + 10 \sin(\pi S/2) \, e^{-16S^2}$$

3. RESULTS AND DISCUSSIONS

To enrich the comparison, the results of three well-known versions of the time-varying accelerated coefficient PSO were included; PSOTAC in [27], PSOM in [29], and PSOI in [30]. These algorithms and the algorithms mentioned in section 3 were tested using the I-V data reported in [31-32] for 57 mm diameter commercial silicon solar cell (R.T.C France) under an irradiation of 1000 W/m2 and a temperature of 33 oC. The implementation was conducted in Matlab environment. The simulation was conducted 1000 times, each time it lasted for 1000 iterations. The simulation was run for 20, 50, 100, 500, 750 and 1000 particles/Agents. A comprehensive comparison is made in terms of present RMSE, MAE of the final iteration, the CR to reach RMSE of 1.25% and 2.5%, and the required simulation time. The best ten results arranged in ascending order are shown in Table 2. Figure 3 shows the performance of all PSOAGs at population of 250 agents. The CR is shown in Figure 4 for all the algorithms at three different agents' number; 20, 250 and 1000 agents. The best candidate was found to be PSOAG8 as it has the fastest CR with minimum MAE and one of the minimum RMSE. Figure 5 shows the results of PSOAG8 at different populations; 20, 250 and 1000 agents.

Increasing the agent/particle number improves the performance as expected from metaheuristic techniques. This can be drawn from comparing the performances at 20 agents to 1000 agents. The RMSE and MAE are reduced by 70-87% when the population increased. The same was observed for the CRs in Figure 4 and Table 2, as they were reduced by (up to) 32% and 55% for the convergence to RMSE with values of 0.01 and 0.02, respectively. Moreover, the stability of the method was improved as well. Based on Figure 5, each algorithm gave almost the same results when it was repeated 1000 times with population of 1000 agents; the variation box became smaller in Figure 5(c) compared to Figure 5(a)-(b). On the other hand, the computational time became significantly large; e.g. for 1000 agents, the required time was almost 28-41 times the required time for 20 agents and 17-30 times the required time for 250 agents.

The results showed that dividing the population into groups with different social behaviors improved the performance. According to [27, 29-36], some applications required certain symmetric social functions, while others required asymmetric social functions. By dividing the population into groups, each group could be assigned to a different function. Mixing these functions for the same application improved the algorithm performance in terms of stability, CR, RMSE, and MAE. That was because the application could rely on its preferred function.

The results of Table 2 showed that best results for modeling the data listed in [31-32] could be obtained using autonomous groups with symmetric and asymmetric power functions as social learning coefficients. This was observed by tracking the best performance, which was achieved by PSOAG3 and PSOAG8. These two methods showed similar performances in terms of RMSE and MAE when the number of populations was high; i.e. above 500 agents. As the population's size decreased, PSOAG8 became better than PSOAG3, where the RMSE and MAE of PSOAG8 were 13.6% and 13.9% smaller than their corresponding values for PSOAG3, respectively. On the other hand, PSOAG8 had a CR that was 2-20% faster than PSOAG3 in all tests except for the test with population size of 250, where it was slower by up to 5.5%.

Therefore, if computational time is not important, then PSOAG8 and PSOAG3 with 1000 agents are the best candidates. However, if the computational time is important as well as the error, then PSOAG8 with 250 agents is the best candidate. The choice of 250 agents is illustrated in Figure 5. Without losing generality, PSOAG8 gives the best results compared to other techniques; e.g. Figure 3 shows the superiority of PSOAG8 among the others for 250 agents. Therefore, it was selected in this work.

Table 2. Comparison between all PSO versions in ascending order (only the first 10)

| RMSE IN FINAL SOL			MAE IN FINAL SOL			CONV <0.02			CONV <0.01		
	Agent	Mean		Agent	Mean		Agent	Mean		Agent	Mean
PSOAG3	1000	0.0132	PSOAG8	1000	0.011	PSOAG8	1000	406.3	PSOAG8	1000	662.6
PSOAG8	1000	0.0135	PSOAG3	1000	0.011	PSOAG8	750	453.8	PSOAG8	750	705.6
PSOAG8	750	0.0147	PSOAG8	750	0.012	PSOAG3	1000	487.8	PSO	1000	750.0
PSOAG3	750	0.0147	PSOAG3	750	0.012	PSOAG7	1000	497.7	PSOAG8	500	757.7
PSOAG3	500	0.0163	PSOAG3	500	0.014	PSOAG2	1000	510.2	PSO	750	770.7
PSOAG8	500	0.0169	PSOAG8	500	0.014	PSOTAC	1000	516.4	PSOAG3	1000	776.6
PSOAG2	1000	0.0174	PSOAG2	1000	0.015	PSOAG8	500	526.3	PSOAG3	750	805.0
PSOAG7	1000	0.0181	PSOAG7	1000	0.015	PSOAG3	750	537.8	PSOTAC	1000	814.0
PSOTAC	1000	0.0188	PSOTAC	1000	0.016	PSOAG7	750	546.1	PSOAG2	1000	819.7
PSOAG8	250	0.0200	PSOAG8	250	0.017	PSOTAC	750	548.1	PSOAG8	250	823.4
PSOAG7	750	0.0205	PSOAG7	750	0.017	PSOAG2	750	566.5	PSO	500	824.7
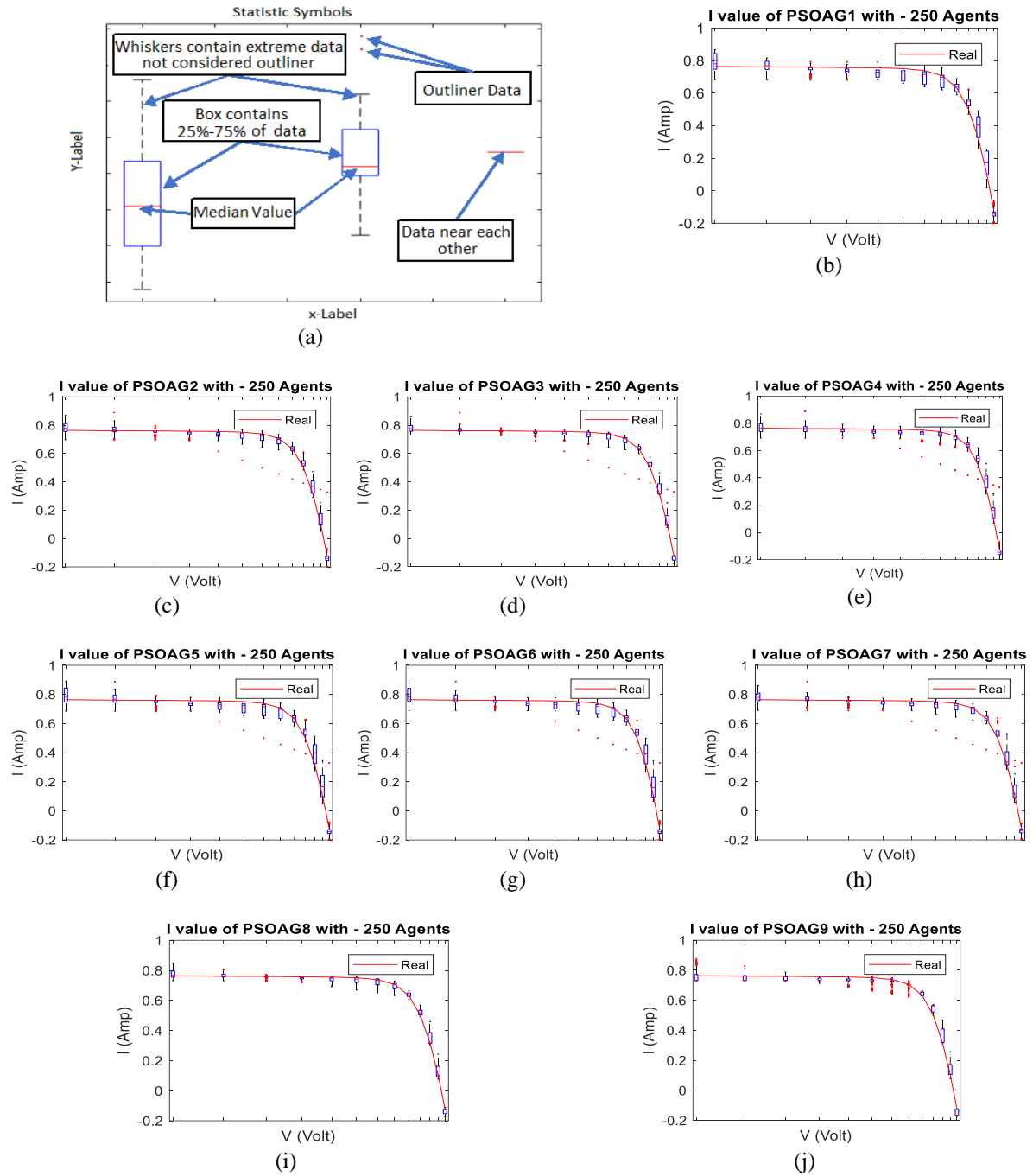
Figure 3. Comparison between PSO versions at 250 agents, (a) Key Figure to represent the statistical analysis graphically, (b) PSOAG1, (c) PSOAG2, (d) PSOAG3, (e) PSOAG4, (f) PSOAG5, (g) PSOAG6, (h) PSOAG7, (i) PSOAG8, and (j) PSOAG9
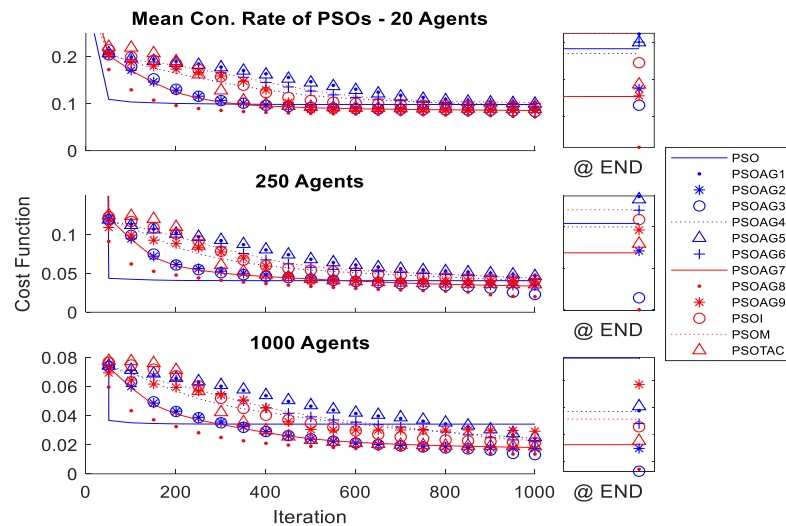
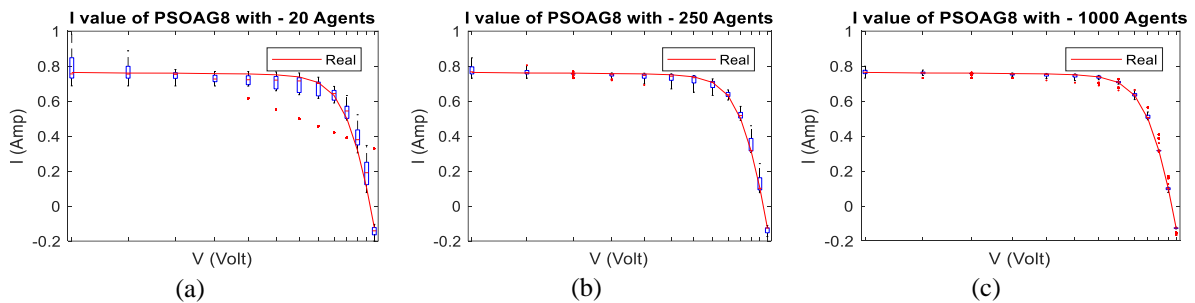Figure 4. Convergence rate of all PSO versions for 20, 250, and 1000 Agents

Figure 5. Comparison between PSOAG8 at different agents' number, (a) 20, (b) 250, and (c) 1000 Agents

4.   CONCLUSION

In this paper, several modified versions of autonomous grouping PSO were proposed. These versions were inspired by the diversity of intellectual ability for individuals in their colonies. To evaluate the performance of the algorithms, the I-V data was used. The results were compared in terms of RMSE, MAE, CR to RMSE values of 0.02 and 0.01, and the simulation time. For better comparisons, these results were compared to newly developed methods; i.e. PSOTAC, PSOM and PSOI. The results showed that PSOAG8 was the best candidate as it had the fastest CR, and one of the minimum RMSE and MAE compared to other algorithms. It gave better performance in terms of RMSE and MAE compared to the autonomous particles groups for PSO when population was of size 500 or less. These values decreased by up to 14%. Moreover, the proposed algorithm showed up to 20% faster CR compared to the autonomous particles groups for PSO algorithm. More functions will be explored in the future for the social coefficients' behavior.

REFERENCES

[1]   M. Kermadi and E. M. Berkouk, "Artificial intelligence-based maximum power point tracking controllers for Photovoltaic systems: Comparative study," *Renew. Sustain. Energy Rev.*, vol. 69, pp. 369-386, 2017. https://doi.org/10.1016/j.rser.2016.11.125.

[2]   R. C. M. Gomes, M. A. Vitorino, M. B. de Rossiter Correa, D. A. Fernandes, and R. Wang, "Shuffled Complex Evolution on Photovoltaic Parameter Extraction: A Comparative Analysis," *IEEE Trans. Sustain. Energy*, vol. 8, no. 2, pp. 805-815, Apr. 2017. DOI: 10.1109/TSTE.2016.2620941.

[3]   V. J. Chin, Z. Salam, and K. Ishaque, "Cell modelling and model parameters estimation techniques for photovoltaic simulator application: A review," *Appl. Energy*, vol. 154, pp. 500-519, 2015. https://doi.org/10.1016/j.apenergy.2015.05.035.

[4]   R. Abbassi, A. Abbassi, M. Jemli, and S. Chebbi, "Identification of unknown parameters of solar cell models: A comprehensive overview of available approaches," *Renew. Sustain. Energy Rev.*, vol. 90, pp. 453-474, 2018.

https://doi.org/10.1016/j.rser.2018.03.011.

[5] O. Mares, M. Paulescu, and V. Badescu, "A simple but accurate procedure for solving the five-parameter model," *Energy Convers. Manag.*, vol. 105, pp. 139-148, 2015. https://doi.org/10.1016/j.enconman.2015.07.046.

[6] K. Ishaque, Z. Salam, and H. Taheri, "Simple, fast and accurate two-diode model for photovoltaic modules," *Sol. Energy Mater. Sol. Cells*, vol. 95, no. 2, pp. 586-594, 2011. https://doi.org/10.1016/j.solmat.2010.09.023.

[7] V. Khanna, B. K. Das, D. Bisht, Vandana, and P. K. Singh, "A three diode model for industrial solar cells and estimation of solar cell parameters using PSO algorithm," *Renew. Energy*, vol. 78, pp. 105-113, 2015. https://doi.org/10.1016/j.renene.2014.12.072.

[8] J. Ma, Z. Bi, T. O. Ting, S. Hao, and W. Hao, "Comparative performance on photovoltaic model parameter identification via bio-inspired algorithms," *Sol. Energy*, vol. 132, pp. 606-616, 2016. https://doi.org/10.1016/j.solener.2016.03.033.

[9] S. Pindado and J. Cubas, "Simple mathematical approach to solar cell/panel behavior based on datasheet information," *Renew. Energy*, vol. 103, pp. 729-738, 2017. https://doi.org/10.1016/j.renene.2016.11.007.

[10] A. N. Celik and N. Acikgoz, "Modelling and experimental verification of the operating current of mono-crystalline photovoltaic modules using four- and five-parameter models," *Appl. Energy*, vol. 84, no. 1, pp. 1-15, 2007. https://doi.org/10.1016/j.apenergy.2006.04.007.

[11] A. R. Jordehi, "Parameter estimation of solar photovoltaic (PV) cells: A review," *Renew. Sustain. Energy Rev.*, vol. 61, pp. 354-371, 2016. https://doi.org/10.1016/j.rser.2016.03.049.

[12] S. Raj, A. Kumar Sinha, and A. K. Panchal, "Solar cell parameters estimation from illuminated I-V characteristic using linear slope equations and Newton-Raphson technique," *J. Renew. Sustain. Energy*, vol. 5, no. 3, 2013. https://doi.org/10.1063/1.4803748.

[13] T. Easwarakhanthan, J. Bottin, I. Bouhouch, and C. Boutrit, "Nonlinear Minimization Algorithm for Determining the Solar Cell Parameters with Microcomputers," *Int. J. Sol. Energy*, vol. 4, no. 1, pp. 1-12, 1986. https://doi.org/10.1080/01425918608909835.

[14] R. Gottschalg, M. Rommel, D. G. Infield, and M. J. Kearney, "The influence of the measurement environment on the accuracy of the extraction of the physical parameters of solar cells," *Meas. Sci. Technol.*, vol. 10, no. 9, pp. 796-804, 1999. DOI:10.1088/0957-0233/10/9/306.

[15] J. Ma, "Optimization Approaches for Parameter Estimation and Maximum Power Point Tracking (MPPT) of Photovoltaic Systems," pp. 26-61, 2014.

[16] M. Ulaganathan and D. Devaraj, "Real coded genetic algorithm for optimal parameter estimation of solar photovoltaic model," *1st Int. Conf. Emerg. Trends Eng. Technol. Sci. ICETETS 2016 - Proc.*, 2016. DOI: 10.1109/ICETETS.2016.7603085.

[17] P. Rajkumar and D. Devaraj, "A Hybrid Particle Swarm Optimisation approach for Reactive Power Optimisation in electricity market," *Int. J. Energy Technol. Policy*, vol. 7, no. 4, pp. 379-392, 2011. DOI: 10.1504/IJETP.2011.039220.

[18] A. Askarzadeh and A. Rezazadeh, "Parameter identification for solar cell models using harmony search-based algorithms," *Sol. Energy*, vol. 86, no. 11, pp. 3241-3249, 2012. https://doi.org/10.1016/j.solener.2012.08.018.

[19] C. Chellaswamy and R. Ramesh, "Parameter extraction of solar cell models based on adaptive differential evolution algorithm," *Renew. Energy*, vol. 97, pp. 823-837, 2016. https://doi.org/10.1016/j.renene.2016.06.024.

[20] L. Guo, Z. Meng, Y. Sun, and L. Wang, "Parameter identification and sensitivity analysis of solar cell models with cat swarm optimization algorithm," *Energy Convers. Manag.*, vol. 108, pp. 520-528, 2016. https://doi.org/10.1016/j.enconman.2015.11.041

[21] M. A. Awadallah, "Variations of the bacterial foraging algorithm for the extraction of PV module parameters from nameplate data," *Energy Convers. Manag.*, vol. 113, pp. 312-320, 2016. https://doi.org/10.1016/j.enconman.2016.01.071.

[22] Q. Niu, L. Zhang, and K. Li, "A biogeography-based optimization algorithm with mutation strategies for model parameter estimation of solar and fuel cells," *Energy Convers. Manag.*, vol. 86, pp. 1173-1185, 2014. https://doi.org/10.1016/j.enconman.2014.06.026.

[23] J. P. Ram, T. S. Babu, T. Dragicevic, and N. Rajasekar, "A new hybrid bee pollinator flower pollination algorithm for solar PV parameter estimation," *Energy Convers. Manag.*, vol. 135, pp. 463-476, 2017. https://doi.org/10.1016/j.enconman.2016.12.082.

[24] D. Oliva, A. A. Ewees, M. A. El Aziz, A. E. Hassanien, and M. Peréz-Cisneros, "A Chaotic Improved Artificial Bee Colony for Parameter Estimation of Photovoltaic Cells," *Energies*, vol. 10, no. 7, p. 865, 2017. DOI: 10.3390/en10070865.

[25] Y. Zhang, S. Wang, and G. Ji, "A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications," *Math. Probl. Eng.*, vol. 1, pp. 1-38, 2015. https://doi.org/10.1155/2015/931256.

[26] D. Oliva, M. Abd El Aziz, and A. Ella Hassanien, "Parameter estimation of photovoltaic cells using an improved chaotic whale optimization algorithm," *Appl. Energy*, vol. 200, pp. 141-154, 2017. https://doi.org/10.1016/j.apenergy.2017.05.029.

[27] A. R. Jordehi, "Time varying acceleration coefficients particle swarm optimisation (TVACPSO): A new optimisation algorithm for estimating parameters of PV cells and modules," *Energy Convers. Manag.*, vol. 129, pp. 262-274, 2016.

[28] Z. Tang and D. Zhang, "A modified particle swarm optimization with an adaptive acceleration coefficients," *Proc. - 2009 Asia-Pacific Conf. Inf. Process. APCIP 2009*, vol. 2, pp. 330-332, 2009. DOI: 10.1109/APCIP.2009.217

[29] G. Q. Bao and K. F. Mao, "Particle swarm optimization algorithm with asymmetric time varying acceleration coefficients," *2009 IEEE Int. Conf. Robot. Biomimetics, ROBIO 2009*, no. 3, pp. 2134-2139, 2009. DOI:

10.1109/ROBIO.2009.5420504.

[30] Z. Cui, J. Zeng, and Y. Yin, "An improved PSO with time-varying accelerator coefficients," *Proc. - 8th Int. Conf. Intell. Syst. Des. Appl. ISDA 2008*, vol. 2, pp. 638-643, 2008. DOI: 10.1109/ISDA.2008.86.

[31] S. Mirjalili, A. Lewis, and A. S. Sadiq, "Autonomous Particles Groups for Particle Swarm Optimization," *Arab. J. Sci. Eng.*, vol. 39, no. 6, pp. 4683-4697, 2014. https://doi.org/10.1007/s13369-014-1156-x.

[32] M. R. AlRashidi, M. F. AlHajri, K. M. El-Naggar, and A. K. Al-Othman, "A new estimation approach for determining the I-V characteristics of solar cells," *Sol. Energy*, vol. 85, no. 7, pp. 1543-1550, 2011. DOI: 10.1016/j.solener.2011.04.013.

[33] Z. Tang and D. Zhang, "A modified particle swarm optimization with an adaptive acceleration coefficients," *Proc. - 2009 Asia-Pacific Conf. Inf. Process. APCIP 2009*, vol. 2, pp. 330-332, 2009. DOI: 10.1109/APCIP.2009.217.

[34] M. Alshabi, C. Ghenai, M. Bettayeb, "Modified Asymmetric Time-varying Coefficient of Particle Swarm Optimization," in *2020 Advances in Science and Engineering Technology (ASET) International Conferences*, 2020. DOI: 10.1109/ASET48392.2020.9118366.

[35] M. Alshabi, C. Ghenai, M. Bettayeb, "Sinusoidal Asymmetric Time-varying Coefficient of Particle Swarm Optimization," in *2020 International Conference on Communications, Signal Processing, and their Applications (ICCSPA)*, 2020.

[36] M. Alshabi, C. Ghenai, M. Bettayeb, "Improved Asymmetric Time-varying Coefficient of Particle Swarm Optimization," in *2020 IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*, 2020.