Review and comparison of BAT and PSO MPPT's based algorithms for photovoltaic system

HAITHEM CHAIEB
Unité de Recherche Etude de Systèmes Industriels et Energies Renouvelables (ESIER)
Ecole Nationale d'Ingénieurs de Monastir, Avenue Ibn El Jazzar, 5019 Monastir, TUNISIA
h.chaieb01@gmail.com
ANIS SAKLY
Unité de Recherche Etude de Systèmes Industriels et Energies Renouvelables (ESIER)
Ecole Nationale d'Ingénieurs de Monastir, Avenue Ibn El Jazzar, 5019 Monastir, TUNISIA

Abstract:-Optimization can be defined as an effort of generating solutions to a problem under constrained circumstances utilizing the existing resources with the finest way. Literature proposes many variants of optimization methods inspired from nature such as ev olutionary algorithms and swarms intelligence algorithms. Swarm intelligence is considered as a powerful method suitable for optimization problems. For instance, Bat algorithm has been inspired by bats behavior during their flight and hunting. Particle Swarm Optimization algorithm has been inspired by social behavior of fish schooling or bird flocking

In this work, an attempt is made to validate experimentally and analyses the performance of Bat and PSO algorithm. A standard test has been carried to evaluate the ability to track the global maximum point of the photovoltaic panel under a step change in irradiance Furthermore under partial shading conditions. The main equations that govern the behavior of each algorithm are also explained. Furthermore, this paper outlines how far BA algorithm outperforms PSO with respect to the tracking capability, transient behavior, and convergence criteria as results of his excellent features. It is envisaged that the BA algorithm can be suitably used as an MPPT technique, particularly for large PV array under various weather condition.

Key-Words: - MPPT; swarm intelligence; Bat algorithm; partially shading condition; PSO.

1 Introduction

The production of energy is a major challenge for the coming years. Indeed, the energy need for industrialised societies is increasing from day to others. Solar energy is a direct conversion of light emitted solar cell by the sun into electrical energy. The PV panel, or photovoltaic generator, is itself constituted of an association of series and parallel of a necessary number of PV modules to assure the required energy [1, 2].

Due to the continuous variation of environmental conditions mainly of temperature and solar irradiance, the PV panel characteristic presents a non-linear curve with maximum power point varying over time [3]. The MPPT objective is to improve and optimise the use of photovoltaic systems. Therefore, maximise the array efficiency in order to guarantee maximum efficiency [4, 5].

Specific methods exist to optimise power from the solar panel and bring it to operate at his maximum points as their specifications without knowing these points in advance, and without knowing when they were changed or what the reasons are. The literature proposes multiple choices of optimisation methods such as conventional and non-conventional algorithms that are mainly inspired by nature [9, 13].

Biologically inspired, so called non-conventional algorithms have gained much attention due to its ability to handle multi-peaks PV curve, faster convergence and guaranteed convergence to the global peak [5,6]. Mainly there are two concepts developed in bio-inspired computation: such evolutionary algorithms and the swarm intelligence algorithms.

Evolutionary algorithms (EA) are optimisation techniques [7] that base on Darwin's principle of survivor [8]. Indeed, he considers that competent individuals have the greater chances to survive in nature. Genetic algorithms, genetic programming and evolution strategies are some popular disciplines from EA optimisation method.

Swarm intelligence is based on the principle of the collaborative behaviour of natural self-organized systems. It was introduced by Beny in 1989 and was used in a lot of optimisation problem since then [5, 16]. Literature introduces many variants of this type of algorithm. Ant colony systems (ACS) proposed by Dorigo et al. [10] is inspired by the foraging behaviour of ants. Cuckoo search algorithm (CS) introduced by Yang et al. [11] is inspired by the parasitism behaviour of cuckoo species. Cat Swarm Algorithm (CSO) presented by Chu et al. [12] come from observing the behaviour of cats when catching their prey. Bat algorithm (BA) introduced by Yang in 2010 [18] and was inspired by bat behaviour. Particle swarm optimisation (PSO) proposed by Kennedy and Eberhart [17] developed after analysing the social and cognitive behaviour of fish or bird swarms.

In this paper, we are interested in swarm intelligence based optimisation method. BA and PSO algorithm are implemented to extract the maximum power available from the photovoltaic panel. Therefore, an experimental validation of these techniques under the various condition is proposed.

The remainder of the paper is organised as follows. section two, describe BA and PSO algorithms used to perform the review test; the third section describes the experimental test bench, in the fourth section we focus on results under different condition such as, under uniform condition, under partial shading condition (PSC) and with irradiance step change; the main conclusion are made in section five.

2 MPPT methods

2.1. Particle swarm optimization.

Particle swarm optimization is an intelligent optimisation, simple and effective meta-heuristic approach. PSO is a non-conventional, population-based search method based on swarm behaviour in nature. The principle of this algorithm was inspired by the behaviour of bird flocks, to overcome the problems associated with search and optimisation [13, 17].

The PSO algorithm searches the space of an objective function by adjusting the trajectories of individual agents, called particles. Each particle is attracted toward the position of the current global best (G_{best}) and its own best location (P_{best}) in history, while at the same time it has a tendency to move randomly. When a particle finds a location that is better than any previously found locations, updates that location as the new current best for particle "i". There is a current best for all "n" particles at any time "t" during iterations. The aim is to find the global best among all the current best solutions until the objective no longer improves or after a certain number of iterations. The movement of particles is schematically represented in Fig.1 [17].

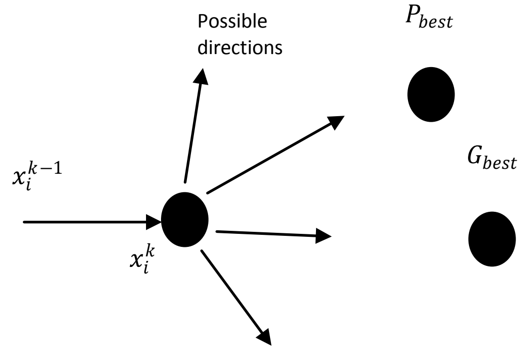

Fig. 1: Motion of Particles in Optimization Process

In this process, each particle represents a possible candidate solution and follows a simple behavior by emulating the success of neighboring particles and its own achieved successes. Therefore, the position of each particle is influenced by the best neighborhood particle P_{best} as well as the global best position found by all particles in the entire population G_{best}.

Indeed, the i eth position x_i for each particle is updated according to the following equation:

$$x_i^{k+1} = x_i^k + \Phi_i^{k+1} \tag{1}$$

"k" represents the iteration counter. The velocity component, Φ_i represents the step size, is also adapted iteratively to render particles capable of potentially visiting any region of the search space. Velocity is adjusted as follows:

$$\Phi_i^{k+1} = w\Phi_i^k + c_1 r_1\{P_{best} - x_i^k\} + c_2 r_2\{G_{best} - x_i^k\} \tag{2}$$

"w" called as the inertia weight that controls the impact of the previous velocity of the particle on its current one. c_1 and c_2 are the acceleration coefficients. r_1 , r_2 are random variables uniformly distributed within [0, 1], $P_{best,i}$ is the personal best position of the particle i, and P_{best} is the best position of the particles in the entire swarm population [14,15].

Consider that the particle position as actual duty cycle and velocity act as the perturbation in the present duty cycle, then the equation can be rewritten as follows.

$$d_i^{k+1} = d_i^k + \Phi_i^{k+1} \quad (3)$$

Indeed, according to (3), resulting perturbation in the present duty cycle depends on P_{best} and G_{best}. The PSO algorithm parameters used in this experiment are presented in the following Table 1.

Table 1. PSO Parameters

Parameters	Value
Population size	5
Dimension number	1
w	0.4
c1	1.2
c2	1.6
r1	0.5
r2	0.7

The proposed flow chart of MPPT based PSO algorithm is shown in below Fig. 2.

The PSO basic operating procedure can be detailed and explained step by step as follows;

Step1: Initialize PSO parameters: swarm size, initial position, initial velocity, and set up iteration counter.

Step2: Evaluate the fitness value of each particle in the swarm.

Step3: Evaluate and update each particle best position P_{best} .

Step4: Evaluate and update global particles best position G_{best} .

Step5: Update Velocity and Position of Each Particle in the swarm

Step6: Check the convergence criterion if met, otherwise, the iteration counter will increase by 1 and go to step 2.

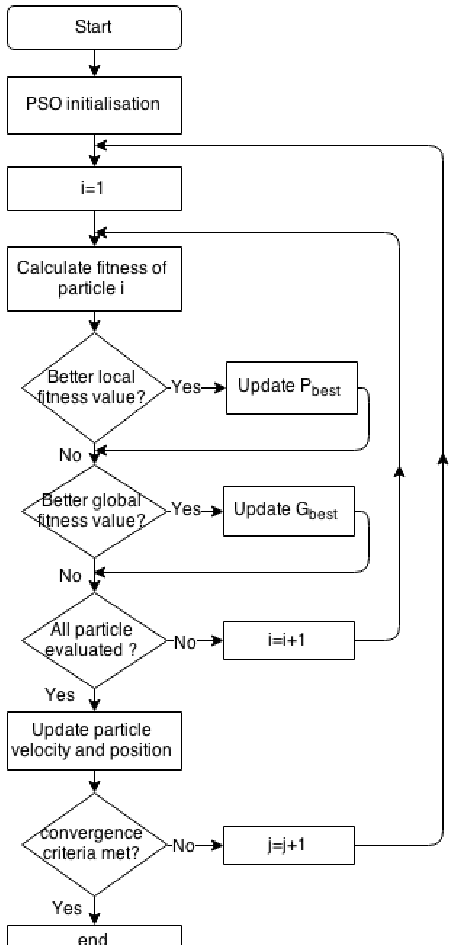

Fig.2: PSO flow chart

2.2. Bat Algorithm.

The BA is a b io-inspired optimization algorithm inspired from natural bats behavior in searching and locating foods using echolocation capability. Bat species emit a t ype of sonar known as e chololocation while hunting for foods, avoid an obstacle while flying even locate their prey and hunting even in complete darkness. These bats emit a very loud sound pulse wave, with signal bandwidth varying with species and often increase by using more harmonics, then listens and analyze for the echo that bounces back from the surrounding objects, [18, 19]. The bat echolocation behavior can be used and formulated in three generalized rules [18].

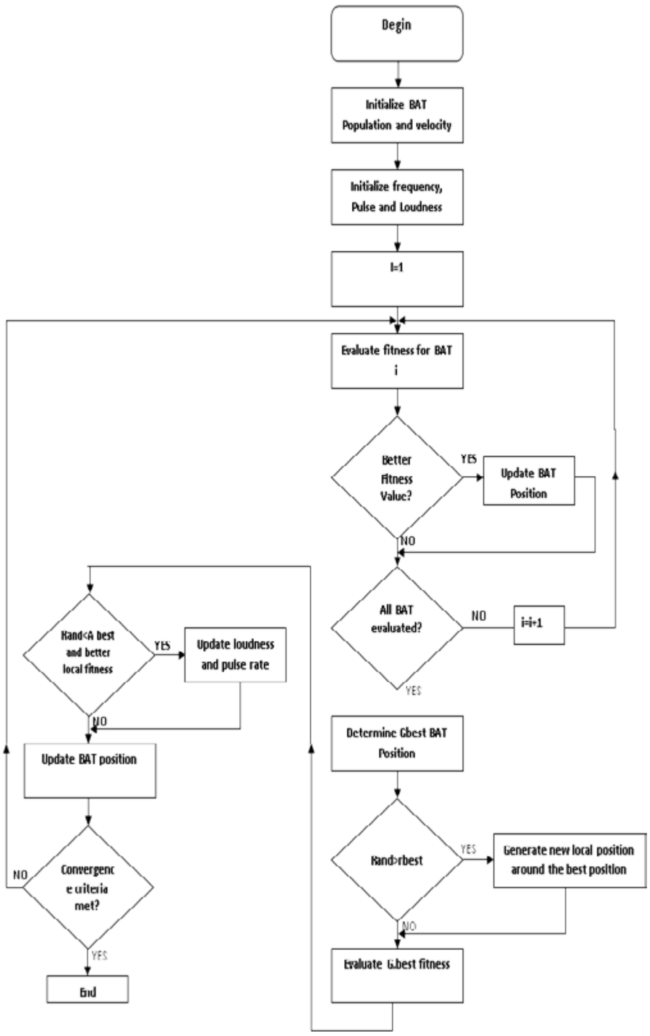

Figure 3: Bat algorithm flowchart

All bats use echolocation to sense distance and differentiates between food and background barriers. Bats fly randomly with a velocity "v_i" at position "x_i" with a f requency "f_i" , varying wavelength, and loudness "A_0" to search for prey. They can automatically adjust the wavelength (or frequency) of their emitted pulses and adjust the pulse rate "r" ∈ [0, 1], depending on t he proximity of their target. Although the loudness can vary in many ways, we assume that the loudness varies from a large (positive) A_0 to a minimum constant value A_{min} .

Search space assumed as a region of many targets, for that the bat algorithm tends to locate the high optimum quality results in the search space. The bats initial population is randomly generated since the final target is unknown at the beginning. Rules should be defined by how their positions and velocities in a d-dimensional search space will be updated [18-21].

Accordingly, the new solutions "x_i^t" and velocities "v_i^t" at a time step "t", which is affected by a randomly predefined frequency "f", are given by following equations:

$$f_i = f_{min} + \beta(f_{max} - f_{min}) \tag{4}$$

$$v_i^t = v_i^{t-1} + \left(x_i^t - x_*\right) f_i \tag{5}$$

$$x_i^t = x_i^{t-1} + v_i^t \tag{6}$$

Where β ∈ [0, 1] indicates a randomly generated vector, "x_*" is the obtained global best location (solution) after comparison of all solutions among "n" bats. "f_i" is a frequency value belonging to the "i" eth bat, "f_{min}" and "f_{max}" are minimum and maximum frequency values, respectively. Initially, each bat is randomly assigned to a frequency that is drawn uniformly from [f_{min} , f_{max}], and "v_i^t" implies the velocity of the "i" eth bat at "t" eth time step.

In order to improve local search capability of the algorithm, once a solution is selected among the current best solutions, a new solution for each bat is generated locally using a random walk.

$$x_{new} = x_{old} + \mathcal{E}\, A^t \tag{7}$$

Where "A^t" represent the average loudness value of all bats at the current time step and "Ɛ" ∈ [-1, 1] is a randomly generated number, while "x_{old}" represents the high-quality solution.

Furthermore, the pulse emission rate r_i and loudness A_i have to be updated accordingly as the iterations proceed and bat gets closer to its final target. Indeed, loudness usually decreases whereas pulse rate emission increases once a bat has found its prey with respect to below equation.

$$A_i^{t+1} = \alpha A_i^t \tag{8}$$

$$r_i^{t+1} = r_i^0(1 - e^{\gamma t}) \tag{9}$$

Where "γ" and "α" are constants, whereas the initial emission rate r_i^t ∈ (0, 1] of the "i" eth bat. Initially, each bat should have different values of loudness and pulse emission rate; this can be achieved by randomization.

Furthermore, the flowchart of the proposed MPPT based bat algorithm is given on Fig.3.

3 System description

In order to assess the performance and analyse the PSO and BA based MPPT algorithms. An experiment was performed during a sunny and clear sky day, in order to obtain a uniform and stable condition. The experimental test bench system is divided into two essential parts: the hardware and the software part as illustrated in Fig.4 (a).

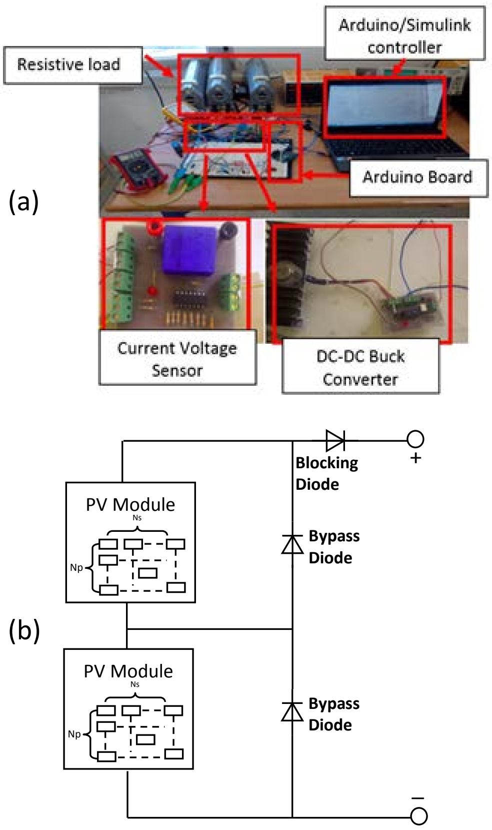

Fig.4: (a) Experimental setup, (b) PV Panel,

The hardware part is constituted with a PV panel, current and voltage sensor and a dc-dc buck converter tuned from Arduino Uno pulse width modulation (PWM) output pin. A variable resistance is connected to the system as a load.

The photovoltaic panel is internally divided into two separate module and consists of 2x33 photovoltaic cells each, connected in series /parallels. Furthermore, two bypasses and one blocking diode are employed as illustrated in Fig.1 (b). The PV panel key parameters are provided in the following Table2.

The software part contains blocks from a MATLAB / Simulink environment with Arduino support package block sets. The program under Simulink includes the block for I-V acquisition as an analogue input and a port initialization for the Arduino Uno board. The duty cycle (Dcy) of used standard buck converter is computed using the proposed MPPT algorithms with Simulink library. The final block has the role in generating the PWM signal command thru Arduino Uno board.

The Arduino UNO based Atmega328p 8 bits AVR RISC-based microcontroller integrate 10 bits analogue to digital converter and six 8 bits PWM channels. Arduino PWM feature is exploited to control the buck converter directly. PWM signal is varying from “0” to “255” which represent a duty cycle value between 0 and 1. Simulink environment with Arduino add-on block-Set allows hardware communication in order to manipulate data and command the buck converter with real time processing.

Table 2. The PV panel specification

Parameters	Label	Value
Maximum power	P_{max}	158W
Rated voltage	V_{MPP}	33.4V
Rated current	I_{MPP}	5 A
Short circuit current	I_{SC}	5.18 A
Open circuit voltage	V_{OC}	41.7 V
V_{oc} coef. of temperature	K_v	-0.13 V/°C
I_{sc} coef. of temperature	K_i	2.5 e-3 A /°C
Module number		2
Cell per module		33cells

4 Experimental validation

In order to validate experimentally the BA algorithm, five bats are initialized randomly. Sampling time is set to “Te = 0.2 seconds". Indeed, every "0.2 seconds” a Dcy update will be sent to the buck converter thru Arduino board. Voltage and current sensor transfer data to be processed thru Arduino analogue input pin for evaluation in the same time step.

The Table 3, present the bats sequence processing in order to compute the duty cycle to be sent to the converter. Bats are presented as Xns, where, "n" is the number of the bat in use (from 1 to 5) and “s” is the next step for current bat evaluation. The first step takes “5Te” (1 second), time for the five bat sent sequentially to the converter and respective power is calculated. The computing time for this first step will end with evaluation for the global best bat.

The next iteration is matching to the first bat with a second evaluation (X21) which correspond to sampling time equal to “6Te”. Accordingly, the best first bat position is computed and a new global position is updated sequentially. The global best position will consider the last five sampling time from “2Te” to “6Te” (bats X12 to X21). Indeed, after passing the first step a new update on g lobal best position is performed every 20 m illiseconds which allow a co nsiderable time to save on algorithm processing.

Regarding the PSO algorithm, the same procedure and computing process as for BA algorithm is employed with little difference on algorithms equation. Five particles were also randomly selected at the start up with parameter as presented in Table 1.

4.1. Uniform irradiation condition

In this case, the temperature and irradiance are maintained constant since they are performed at the same lapse of time. By sweeping duty cycle from 0 to 1, the electrical specifications for the PV panel vary accordingly forming a nonlinear curve “Fig.5”. Indeed the maximum power that can be extracted is 112 watts matching to around 30 volts.

The test under the uniform condition is performed in order to determine the tracking speed beginning from the start up point. The experimentally plotted P-V curves are shown in Fig.6.

Hence, both BA and PSO algorithm are converging to the MPP with 98.2% efficiency. However, BA takes only 3.2 seconds for global convergence and PSO algorithm takes 6.2 seconds.

The fluctuation in power at the start-up is due to randomly picked samples at the beginning of the algorithm. However, the power loss related to this fluctuation can be ignored since the samples get nearer to the MPP in very quick succession.

4.2. Partially shading condition.

The shading condition is caused due to many phenomena such as shadows cast by buildings, tree leaves, and passing clouds etc.… If one of the PV modules is shaded, it acts as a load instead of a power source. In long term conditions, the shaded PV module will be damaged due to hotspots phenomenon. Hence, the bypass diodes are added to protect the PV modules from self-heating during partial shading conditions as shown in below Fig.7 [22, 23].

In the case of uniform irradiation, the diodes are reverse biased and have no effect on the electrical circuit. When the PV module is exposed to a shaded condition, the bypass diodes through this PV module are forward biased and the current passes through the diode set in parallel with the embedded module. Indeed, due to those bypass diode, the P-V curves become more complicated and characterised by multiple peaks instead of a single maximum power point. The maximum power point tracking with the partial shading condition becomes more complicated since P-V curve are characterised by two peaks having one global power [22-25].

Applying the partial shading on PV module using a tinted transparent glass, as presented in Fig.7, the resulted P-V curve is plotted in Figure 8

BA and PSO algorithms will generate new samples of five particles randomly. Algorithms will begin the search for the global peak using the procedures described in Table 3. Both BA and PSO based MPPT method succeed to track the global MPP with output power around 84 w atts and efficiency closer to 98.8 %. The BA algorithm takes around 3.8 s econds to reach the global maxima compared to PSO, which takes around 6.8 seconds for global convergence.

4.3. Irradiance change condition.

This test is performed in order to determine the ability of each method to track the change of the operation point linked to change in irradiance condition [26]. For that, we begin with putting the photovoltaic panel under uniform conditions. After “74” seconds the solar irradiation suddenly changes and getting down. A new maximum with less power is generated accordingly.

In the case of re-tracking, after irradiance step change using PSO algorithm, it requires approximately “6.2 seconds” to be settled to the new MPP with 93 watts “Fig.9”. The PSO algorithm starts with five random particle position which takes a longer time to find the new position. This randomization produces a large fluctuations in the transient state which causing power losses.

After the irradiation change, the BA algorithm starts searching for the best position using five random bats (bat1–bat5). The algorithm starts to iterate according to the sequence in Table III until finding the new MPP. It takes "3.4" seconds until getting the new operating point with "93" watts “Figure 9”. Same as for PSO, BA algorithm exhibit longer fluctuations in the transient state causing power losses in PV module.

Table 3. BA evaluation Sequence

<table>
<tr><td>1Te</td><td>2Te</td><td>3Te</td><td>4Te</td><td>5Te</td><td>6Te</td><td>7Te</td><td>8Te</td><td>9Te</td><td>10Te</td></tr>
<tr><td>1- X_{11}
2-BAT 1 evaluation
3-equation (4),(5),(6)</td><td>1- X_{12}
2- BAT 2 evaluation
3-equation (4),(5),(6)</td><td>1- X_{13}
2- BAT 3 evaluation
3-equation (4),(5),(6)</td><td>1- X_{14}
2- BAT4 evaluation
3-equation (4),(5),(6)</td><td>1- X_{15}
2-BAT5 evaluation
3-equation (4),(5),(6)</td><td>1- X_{21}
2- BAT1 evaluation
3-equation (4),(5),(6)</td><td>1- X_{22}
2- BAT2 evaluation
3-equation ((4),(5),(6)</td><td>1- X_{23}
2- BAT3 evaluation
3-equation (4),(5),(6)</td><td>1-X_{23}
2- BAT3 evaluation
3-equation (4),(5),(6)</td><td>1- X_{23}
2- BAT3 evaluation
3-equation (4),(5),(6)</td></tr>
<tr><td colspan="5">Global bats evaluation</td><td></td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td colspan="5">Global bats evaluation</td><td></td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td colspan="5">Global bats evaluation</td><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td colspan="5">Global bats evaluation</td><td></td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td colspan="5">Global BATS evaluation</td><td></td></tr>
<tr><td></td><td></td><td></td><td></td><td></td><td colspan="5">Global bats evaluation</td></tr>
</table>

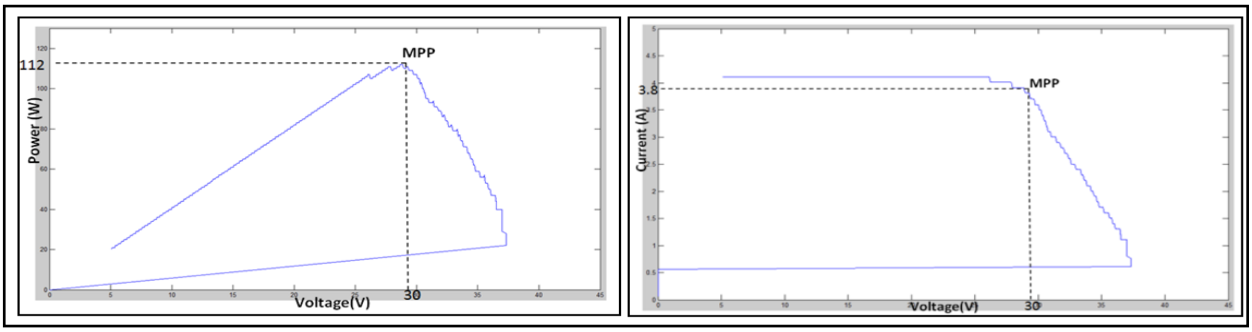

Fig.5: plotted P-V and I-V curve

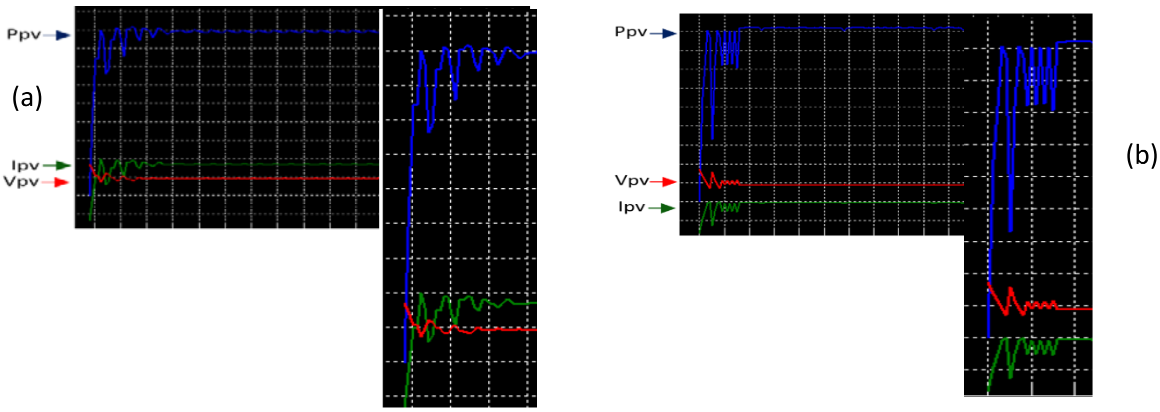

Fig.6: Experimental results for uniform irradiation with (a) PSO and (b) BA methods

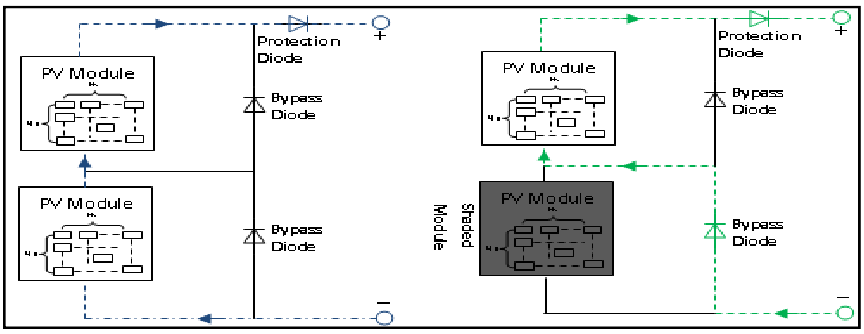

Figure 7: PV panel under PSC

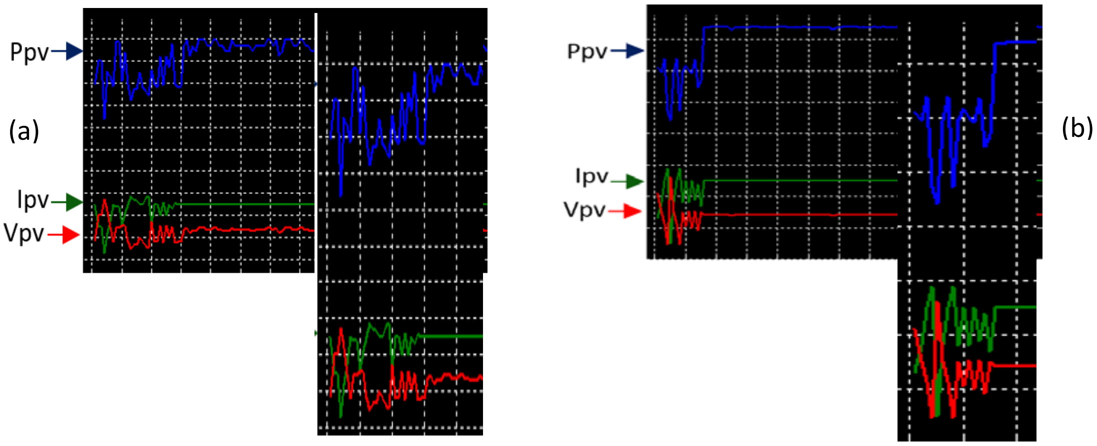

Fig.8: Experimental results for shading condition with (a) PSO and (b) BA methods

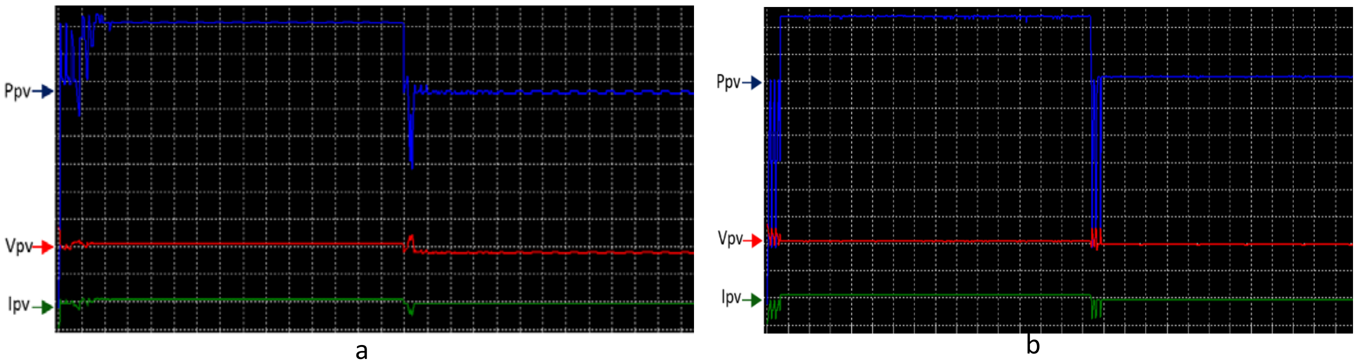

Fig.9: Experimental results for (a) PSO and (b) BA methods under irradiance change condition

5 Conclusion

In this paper, an experimental validation and comparison are performed using BA and PSO based MPPT algorithm. Algorithms are implemented on an Arduino Uno board, with fast and low-cost ADC for current and voltage sampling. Furthermore, Simulink environment is used for real-time monitoring.

Through these experiments, it was concluded that the BA is better than both PSO method in term of tracking speed however, they exhibit the same tracking efficiency. The test is performed under three different condition such as uniform, partially shaded and irradiance change condition.

Experimental results in this paper come with accordance of what is found in literature either theoretically or by simulation. Indeed BA algorithm based on the principle of frequency tuning and the pulse rate changes leads to a good affinity from the ideal position especially in partially shading condition and irradiance change test. Compared to PSO algorithm that has a slow rate of convergence in finding the globally optimal solution compared to BA algorithm.

Acknowledgments
The authors of this work would like to thank the Engineering University of Monastir, laboratory ESIER, and the Ministry of Higher Education and Scientific Research of Tunisia for providing the facilities and financial support to conduct this research.

References

[1] Prospects for Solar and Renewable Power, Nanophysics of Solar and Renewable Energy. (2012) 223–230. doi:10.1002/9783527646289.ch11.

[2] Benemann, J., Chehab, O., Schaar-Gabriel, E.: "Building-integrated PV modules"Solar Energy Materials and Solar Cells, 67, (1-4), pp. 345–354 (2001).

[3] S.Z. Aljoaba, A.M. Cramer, B.L. Walcott, Experimental characterization of a thermoelectrical model of photovoltaic modules, 2012 38 th IEEE Photovoltaic Specialists Conference. (2012). doi:10.1109/pvsc.2012.6318296.

[4] S.K. Sahoo, M. Bansal, MPPT Techniques - A Review, Advanced Materials Research. 1055 (2014) 182–187. doi:10.4028/www.scientific.net/amr.1055.182

[5] E.K.P. Chong, S.H. Zak, An Introduction to Optimization (Wiley-Interscience Series in Discrete Mathematics and Optimization), third ed., Wiley Interscience, 2008.

[6] X. Yang, Nature-Inspired Metaheuristic Algorithms, second ed., Luniver Press, 2010.

[7] Charles Darwin. The origin of species. John Murray, London, UK, 1859.

[8] A.E. Eiben and J.E. Smith. Introduction to Evolutionary Computing. Springer-Verlag, Berlin, 2003.

[9] Liu, G.: "Discussion and Research of Photovoltaic MPPT Methods"The Open Electrical & Electronic Engineering Journal, 7, (1), pp. 123–130 (2013).

[10] M. Dorigo, V. Maniezzo, A. Colorni, Ant system: optimization by a colony of cooperating agents, IEEE Trans. Syst. Man Cybern. B: Cybern. 26 (1) (1996) 29–41.

[11] X.-S. Yang, S. Deb, Cuckoo search via levy flights, in: Nature Biologically Inspired Computing, 2009. N aBIC 2009. World Congress on, 2009, pp. 210–214.

[12] S.-C. Chu, P.-W. Tsai, J.-S. Pan, Cat swarm optimization, in: Q. Yang, G. Webb (Eds.), PRICAI 2006: Trends in Artificial Intelligence, Vol. 4099 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2006, pp. 854–858.

[13] Salam, Z., Ahmed, J., Merugu, B.S.: "The application of soft computing methods for MPPT of PV system: A technological and status review" Applied Energy, 107, pp. 135–148 (2013).

[14] Miyatake, M., Veerachary, M., Toriumi, F., Fujii, N., Ko, H.: "Maximum Power Point Tracking of Multiple Photovoltaic Arrays: A PSO Approach"IEEE Transactions on Aerospace and Electronic Systems, 47, (1), pp. 367–380 (2011).

[15] W. Yunliang, B. Nan, Research of MPPT control method based on PSO algorithm, 2015 4 th International Conference on C omputer Science and Network Technology (ICCSNT). (2015). doi:10.1109/iccsnt.2015.7490840.

[16] RS Parpinelli and HS Lopes. New inspirations in swarm intelligence: a survey. International Journal of Bio-Inspired Computation, 3(1):1–16, 2011

[17] J. Kennedy, R. Eberhart, Particle swarm optimization, in: Neural Networks, 1995. Proceedings. IEEE International Conference on, v ol. 4, 1 995, pp.1942–1948.

[18] X.-S. Yang, A new metaheuristic bat-inspired algorithm, in: J. Gonzlez, D. Pelta,C.Cruz, G. Terrazas, N. rasnogor (Eds.), Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), Vol. 284 of Studies in Computational Intelligence, Springer, Berlin, Heidelberg, 2010, pp. 65–74.

[19] Oshaba, A., Ali, E., Elazim, S.A.: “MPPT control design of PV system supplied SRM using BAT search algorithm”Sustainable Energy, Grids and Networks, 2, pp. 51–60 (2015).
[20] B. Xing, W.-J. Gao, Bat Inspired Algorithms, Innovative Computational Intelligence: A Rough Guide to 134 Clever Algorithms Intelligent Systems Reference Library. (2013) 39–44. doi:10.1007/978-3-319-03404-1_3.
[21] Yılmaz, S., Küçüksille, E.U.: “A new modification approach on bat algorithm for solving optimization problems”Applied Soft Computing, 28, pp. 259–275 (2015).
[22] M.E. Basoglu, B. Cakir, Investigation of solar panel characteristics and MPPT performance under partial shading conditions, 2015 9th International Conference on Electrical and Electronics Engineering (ELECO). (2015). doi:10.1109/eleco.2015.7394633.
[23] B. N. Alajmi, K. H. Ahmed, S. J. Finney, and B. W. Williams, “A maximum power point tracking technique for partially shaded photovoltaic systems in microgrids,” IEEE Trans. Ind. Electron., vol. 60, no. 4, pp. 1596–1606, Apr. 2013.
[24] K. Ishaque, Z. Salam, and H. Taheri, “Syafaruddin, modeling and simulation of photovoltaic (PV) system during partial shading based on a two diode model,” Simul. Model. Practice Theory, vol. 19, no. 7, pp. 1613–1626, 2011.
[25] J. Young-Hyok, J. Doo-Yong, K. Jun-Gu, K. Jae-Hyung, L. Tae-Won, and W. Chung-Yuen, “A real maximum power point tracking method for mismatching compensation in PV array under partially shaded conditions,” IEEE Trans. Power Electron., vol. 26, no. 4, pp. 1001–1009, Apr. 2011.
[26] S.K. Kollimalla, M.K. Mishra, Variable Perturbation Size Adaptive P&O MPPT Algorithm for Sudden Changes in Irradiance, IEEE Transactions on Sustainable Energy. 5 (2014) 718–728. doi:10.1109/tste.2014.2300162.