In this paper, a multi-group salp swarm algorithm (MGSSA) is presented for estimating the photovoltaic (PV) solar cell models. The SSA is a metaheuristic technique that mimics the social behavior of the salp. The salps work in a group that follow a certain leader. The leader approaches the food source and the rest follows it, hence resulting in slow convergence of SSA toward the solution. For several groups, the searching mechanism is going to be improved. In this work, a recently developed algorithm based on several salp groups is implemented to estimate the single-, double-, triple-, Quadruple-, and Quintuple-diode models of a PV solar cell. Six versions of MGSSA algorithms are developed with different chain numbers; one, two, four, six, eight and half number of the salps. The results are compared to the regular particle swarm optimization (PSO) and some of its newly developed forms. The results show that MGSSA has a faster convergence rate, and shorter settling time than SSA. Similar to the inspired actual salp chain, the leader is the most important member in the chain; the rest has less significant effect on the algorithm. Therefore, it is highly recommended to increase the number of leaders and reduce the chain length. Increasing the number of leaders (number of groups) can reduce the root mean squared error (RMSE) and maximum absolute error (MAE) by 50% of its value.

This is an open access article under the CC BY-SA license.

Corresponding Author:
Mohammad Al-Shabi
Department of Mechanical and Nuclear Engineering
University of Sharjah
Sharjah, United Arab Emirates
Email: malshabi@Sharjah.ac.ae

1. INTRODUCTION
The world energy demand (WED) continues to increase as a result of the population and economic growth and the raise in the living standard of human being. Most of the WED is generated from the combustion of fossil fuels. The high greenhouse gas emissions released from burning of fossil fuel are responsible for the air pollution and global warming. Another challenge is the extraction of fossil fuels reserves at higher rate (depletion of natural resources) due to the increase in energy demand. To address all these energy challenges from fossil fuel, sustainable and renewable energy systems are needed. Among alternative sources of energy, solar photovoltaic energy has gained a great attention. PV (photovoltaic) solar system is one of the most used solar devices for direct conversion of solar energy to electrical power due to its silent operation and ecofriendly mature (zero CO₂ emissions). Photovoltaic sources (PVs) have become
the most widely adopted source of renewable energy with 98 GW (Gigawatt of electricity obtained by a PV system) with the new installations in 2017 [1]. The power output and the electrical performance from the PVs depend on environmental factors such as solar irradiance and temperature.

PVs are made of solar PV panels that consists of solar cells combined in a defined manner. Thus, the dynamic behavior of PVs depends upon the basic building block known as solar cell. The datasheet provided by the manufacturer is usually at standard test conditions (STC), which hardly are met in real environmental conditions. Although the provided data is crucial, it is not enough to accurately predict the I-V curve under varying solar temperature and irradiance. The parametric values are also varying with time due to material degradation [2]. Therefore, the method employed to predict the behavioral parameters have gained significant importance in the modelling of PVs [3]. Several modelling techniques are established in literature to exhibit the solar cell behavior. The equivalent electronic circuit models are the most used [4]. Among the equivalent circuit models, the single-diode model (SDM) is considered as the reference model due to its wide acceptance and balanced trade-off between accuracy and complexity [5]. The SDM has five key parameters to describe the non-linearity of the solar cell. The other models include the ideal cell model [6], the double-diode (DDM) [7], and the multi-diode (MDM) [8], which have more complexity relations and calculations.

Several analytical methods and numerical techniques are applied to determine the key parameters of the SDM. The analytical methods are based on data provided by the solar PV manufacturer in datasheet [9] while the numerical methods have used iterative calculation or optimization techniques [10]. Iterative calculation techniques largely depend on initial conditions and have large tendency to converge on local minima and maxima. The large number of iterations also increases the computational cost. In the optimization techniques, the problem of the extraction of the key parameters is converted into an optimization problem [10]. The main target of any optimization technique is to design the cost function that is used to minimize the error between the predicted and the actual or measured data [11]. Metaheuristics techniques have become popular optimization solving methods due to their flexibility and robustness. The meta-heuristic techniques are further classified to evolutionary and swarm intelligence algorithms. Evolutionary algorithms are based on natural evolution patterns. The well-known genetic algorithm is an example of this class which has adopted the evolution theory [12]. In genetic algorithm, a set of random solutions are proposed and are evaluated based on specified control objective functions. Several other algorithms such as differential evolution [13], evolutionary programming and bio-inspired optimization [14] are also known evolutionary algorithms. Contrarily, swarm intelligence techniques are based on intelligence pattern of swarm, herds or flocks in nature. The group behavior of individuals is the foundation of these algorithms. Ant colony optimization [15], artificial bee colony [15], whale optimization [16], firefly optimization [17], particle swarm optimization (PSO) [18]-[21], cuckoo search [22], grey wolf optimization [23] and harmony search [24] represent the swarm intelligence algorithms. Besides of all the merits of metaheuristic algorithms, no individual algorithm is a best fit for all the optimization problems.

Salp swarm algorithm (SSA) is a metaheuristic technique that was developed by Mirjalili et al. in [25] for solving engineering optimization problems. SSA has been inspired from the movements of salps during the navigation and foraging in deep oceans. Salps establish long swarm chain called salp chain. In salp chains, the first salp is the leader and the remaining salps are known as the followers. The leader has the responsibility for the direction and locomotion of the whole chain. This behavior enables the salps to gain swift motion towards the food source [25]. Salp chain formation overcomes the problem of local optima but cannot always perform well in real conditions. Occasionally, the algorithm fails to maintain the balance between exploration and exploitation phases and results in false solution. The best optimization algorithm has the characteristics of avoiding local optima, fast convergence rate (CR) and less complexity [25]. Researchers have adopted SSA in different field such as emission estimation of CO2 [26], for feature sections [10], chemical compound activities [27], power system stabilizer [28], fraction order PID controller [29] and PID-fuzzy controller [30].

In the current study, the SSA is adopted to extract the key parameters of SDM and MDM for the characteristic IV-Curve of a solar cell. A brief comparison is done to show the effectiveness of the proposed algorithm. The paper organization is: section 2 describes the SSA and multi-group SSA (MGSSA), section 3 represents the research methods, section 4 presents the results and discussion, and section 5 includes the conclusions. The main contributions of this work can be briefed as:

- Study the effect of the leaders on the performance of SSA. Then, use the MGSSA with several group levels to optimize an equation. According to the authors’ knowledge, this never done before.
- Derive SDM and MDM models. These includes DDM, triple (TDM), Quadruple (QuD), and Quintuple (QuTd) models. According to the authors’ knowledge, only SDM, DDM, and TDM models were studied in the literature.
- Find the parameters of SDM and MDM PV models from experimental data that is reported in [31].
Study the effect of the model complexity on the performance.

Comprehensive comparison between models and methods is conducted including the root mean squared error (RMSE), maximum absolute error (MAE), CR, min-max-mean of the iterations (MMM), and how many times the method reached the final value (RFV).

The core objective of this study is to examine a newly developed algorithm in improving the accuracy of parametric estimation, and to estimate the parameters of the SDM and MDM modules in order to improve the maximum power point tracking and the results of the simulations of solar PV systems.

Several models were proposed to mimic the PV solar cell (PVSC) behavior. The most famous representations in the literature are the SDM and MDM [32], [33], shown in Figure 1 and Figure 2, respectively. These PVSC models are characterized by (1):

\[
I = I_{ph} - \sum_{i=1}^{m} I_{d_i} \left( e^{\frac{q(V+IR)}{n_i kT}} - 1 \right) - \frac{V+IR}{R_{sh}}
\]

(1)

Where

- \(I_{ph}\), \(I_d\), \(I_{d_i}\) and \(I\) are the Photo-, diode, diode \(i\), and PVSC currents (A), respectively.
- \(q\), \(k\), \(m\) are the Electron charge, Boltzmann’s Constant and number of diodes, respectively. \(m = 1\) for SDM.
- \(R_s, R_{sh}\) are the Series and Shunt Resistances (\(\Omega\)), respectively.
- \(n, n_i\) are the Diode and Diode \(i\), Ideality Factor.
- \(T, V\) are the PVSC Temperature (\(K\)) and Voltage of the PVSC (V), respectively.

The parameters \(I_{ph}, I_d, I_{d_i}\) of the models and \(n, n_i\) are the unknown parameters that need to be obtained based on the I-V curve as the one in [34]. These parameters are required to determine the cell behavior, and hence, needed for the determination of the maximum power. Several works have been conducted in [3], [4], [9], [32]-[41] to find these parameters. The salp swarm optimization (SSA) and several newly developed methods of PSO such as the time-varying accelerated PSO (PSOTAC) [42], modified PSO (PSOM) [43], and the improved PSO (POSI) [44] are used to obtain the above-mentioned parameters. The goal of the optimization method is to minimize the cost function, \(J\), in order to obtain correct values for the parameters. \(J\) is defined, for \(N\) data points, as

\[
J = \frac{1}{N} \sum_{k=1}^{N} \left( I - I_{ph} + \sum_{i=1}^{m} I_{d_i} \left( e^{\frac{q(V+IR)}{n_i kT}} - 1 \right) + \frac{V+IR}{R_{sh}} \right)^2
\]

(2)

2. PROPOSED ALGORITHM
2.1. Introduction- salp swarm algorithm (SSA)

Salp, shown in Figure 3, belongs to the salpidae family and it has a similar behavior of the jellyfish in terms of movements and body transparency. The salp lives in swarms called salp chain. Some works claimed that the chain helps the salp to achieve better and faster maneuvering and foraging. Similar to particle swarm optimization, it starts with selecting locations in the subspace to be a candidate set for the problem, refers to as food source, and then improves these locations iteratively to reach the food location, hence, improves the performance. The entire process mimics the social behavior of a salp flock trying to find a food source. The algorithm of SSA is illustrated using the pseudo-code as shown in Figure 4.
Int J Artif Intell
ISSN: 2252-8938

Estimating PV models using multi-group salp swarm algorithm (Mohammad Al-Shabi)

Figure 3. Salp body [45]

Figure 4. Pseudo-code for SSA

The salps move within the space trying to reach the food in their location; \( F_i \). Each salp memorizes its position and food source position, which are defined for \( n \) particles as \( x_i = [x_{i1} x_{i2} ... x_{in}] \) and \( F_i = [p_{i1} p_{i2} ... p_{in}] \), respectively. These are initialized randomly, but then they are enhanced by (3) and (4) [25]:

\[
x_i^1 = F_i + c_1 \times \text{randn} \times (u_b - l_b) + l_b
\]

(3)

\[
x_i^{t+1} = 0.5(x_i^{t+1} + x_i^j)
\]

(4)

where

- \( x_i^j \) The \( t \)-th salp position at iteration \( i \).
- \( c_1 \) Learning coefficient of the individual best solutions found, it has a value of \( 2e^{-l/L^2} \), where \( l \) and \( L \) are the current and maximum iteration, respectively.
- \( \text{randn} \) It is a random number in \([-1,1]\).
- \( u_b, l_b \) Parameters upper and lower bounds.

2.2. Proposed algorithm - multi-group salp swarm algorithm

In [46], a novel method was presented. This method was referred to as memetic/multi-group salp swarm algorithm, where several groups/chains tried to find the solution as illustrated in the pseudo-code given in Figure 5. The chains regroup after each conducted iteration based on their evaluated fitness function values (FFs) as shown in Figure 6. The salps with the least FFs become the leaders for the chains. The rest of the salps are distributed alternatively among the chains; one salp to chain 1, one to chain 2, … then coming back to chain 1 and so on.

Figure 5. Pseudo-code for memetic salp swarm algorithm

Figure 6. Regrouping method in memetic SSA [46]

3. RESEARCH METHOD

The effects of the chain number and population size on the MGSSA solution are examined in this section. To do that, the method is tested on several benchmarks before it is applied into PV model problem. Several benchmark functions from [42]-[44], [46] are examined and studied in this section using MGSSA.
with different chain numbers. These functions are summarized in Table 1 with a dimension of 30, and they should be optimized toward a value of zero. The results are obtained for MGSSA with single, double, triple and quadruple groups/chains and few of them are listed in Table 2. Each method runs with populations of 10, 20, 30 or 40 particles and last for 1000 epochs. To remove the randomization issue in initialization, each run is repeated 1000 times. The performance in terms MAE, RMSE, CR, and simulation time are recorded. The MGSSA has the same equations as those for SSA. However, they are calculated for each group separately.

### Table 1. Benchmark functions in [42]-[44], [46]

<table>
<thead>
<tr>
<th>Range</th>
<th>Function</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>([-100,100])</td>
<td>(\sum x_i^2)</td>
<td>Unimodal</td>
</tr>
<tr>
<td>([-10,10])</td>
<td>(\sum (</td>
<td>x_i</td>
</tr>
<tr>
<td>([-100,100])</td>
<td>(\max(</td>
<td>x_i</td>
</tr>
<tr>
<td>([-30,30])</td>
<td>(100 \sum [(x_i - x_i^n)^2 + (x_{i-1} - 1)^2])</td>
<td>Unimodal</td>
</tr>
<tr>
<td>([-100,100])</td>
<td>([</td>
<td>x_i</td>
</tr>
<tr>
<td>([-500,500])</td>
<td>(x_i \sin(</td>
<td>x_i</td>
</tr>
<tr>
<td>([-5,12,5,12])</td>
<td>((x_i^2 - 10 \cos(2\pi x_i) + 10))</td>
<td>Multi-modal</td>
</tr>
<tr>
<td>([-32,32])</td>
<td>(-20 \exp \left(-0.2 \frac{\sum(x_i^2)}{\dim}\right) - \exp \left(\frac{1}{\dim} \sum \cos(2\pi x_i)\right) + 20 + \exp(1))</td>
<td>Multi-modal</td>
</tr>
</tbody>
</table>
| \([-50,50]\) | \(0.1 \times \left(\sin(3\pi x_i)^2 + \sum \left[(x_i - 1)^2 (1 + \sin(3\pi x_i)^2)\right] + \left\{\begin{array}{ll} -100(x_i - 5)^4, & x < -5 \\
100(x_i - 5)^4, & x > 5 \end{array}\right.\right)\) | Multi-modal |

Some observations are derived from the results of the benchmarks and they are:

- The RMSE and CR is improved as the number of particles increases.
- The RMSE is improved as the number of leaders and chains increase.

### Table 2. Results of MGSSA at different group numbers for the function 1 of Table 1

<table>
<thead>
<tr>
<th>Number of particles</th>
<th>Number of chains</th>
<th>RMSE</th>
<th>MAE</th>
<th>Simulation time</th>
<th>Converge to RMSE&lt;0.01</th>
<th>Converge to MAE&lt;0.01</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>1</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
</tr>
<tr>
<td>10</td>
<td>2</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
</tr>
<tr>
<td>10</td>
<td>3</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
<td>(\sigma^x)</td>
<td>(\bar{x})</td>
</tr>
</tbody>
</table>

From these results, it is obvious that increasing the number of leaders and populations improves the models’ performance. Consequently, MGSSA is used in this paper to compare between different models of PVSC. In this paper, the conventional SSA method will be tested along with MGSSA with two, four, eight and N/2 chains. These will be referred in this paper as SSA, MGSSA2, MGSSA4, MGSSA8, and MGSSAN, respectively. These are applied to measure the performance of SDM, DDM, QdM, and QM of a PVSC obtained from [31]. For comparison purposes, MGSSA is compared to PSO, PSONTAC [43], PSOM [43], and PSOI [44].

### 4. RESULTS AND DISCUSSIONS

The algorithms mentioned in section 3 were examined using the experimental I-V data of a silicon PVSC reported in [31]. Silicon PVSC is selected because it is currently occupying more than 90% of existing PVSCs. The irradiation was 1000 W/m² and the temperature was 33°C. The implementation was conducted in Matlab environment. The simulation was conducted for SDM and DDM of section 2, each for 1000 times at population of 25, 50, 75, 100, 250 and 500, and the maximum iteration time was set to be 1000 for each. A comparison between the mentioned methods is made in terms of RMSE, and MAE of the final iteration, the CR to reach RMSE of 1.25% and 2.5%, and the total simulation time after setting the simulation environment to be identical; same number of loops and shortcuts. The best results are shown in Table 3. The CR with different number of populations is shown in Figures 7 (a) and 7 (b). Figure 8, including Figures 8(a), 8(b), 8(c), 8(d), and 8(e), and Figure 9, including Figures 9(a) and 9(b), are dedicated to show the effects of number of diodes, and groups, respectively, on the MGSSA performance.
Table 3. Comparison between the best solutions of MGSSA and PSO versions for MDM

<table>
<thead>
<tr>
<th>Method</th>
<th>No. Diode</th>
<th>No. Agent</th>
<th>No. Team</th>
<th>Sim Time (sec)</th>
<th>Iter. to Conv. to 0.01</th>
<th>Iter. to Conv. to 0.02</th>
<th>RMSE (mA)</th>
<th>MAE (mA)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MGSSA</td>
<td>1</td>
<td>100</td>
<td>50</td>
<td>1.988</td>
<td>560</td>
<td>506</td>
<td>2.54</td>
<td>5.45</td>
</tr>
<tr>
<td>MGSSA</td>
<td>5</td>
<td>250</td>
<td>1</td>
<td>2.273</td>
<td>558</td>
<td>512</td>
<td>2.56</td>
<td>5.39</td>
</tr>
<tr>
<td>MGSSA</td>
<td>1</td>
<td>75</td>
<td>37</td>
<td>2.251</td>
<td>565</td>
<td>518</td>
<td>2.56</td>
<td>5.39</td>
</tr>
<tr>
<td>MGSSA</td>
<td>5</td>
<td>100</td>
<td>1</td>
<td>3.068</td>
<td>574</td>
<td>527</td>
<td>2.57</td>
<td>5.41</td>
</tr>
<tr>
<td>MGSSA</td>
<td>1</td>
<td>1000</td>
<td>1</td>
<td>1.06</td>
<td>750</td>
<td>744</td>
<td>34</td>
<td>28</td>
</tr>
<tr>
<td>PSOTAC</td>
<td>1</td>
<td>1000</td>
<td>1</td>
<td>15.9</td>
<td>814</td>
<td>516</td>
<td>19</td>
<td>16</td>
</tr>
<tr>
<td>PSOI</td>
<td>1</td>
<td>1000</td>
<td>1</td>
<td>16</td>
<td>915</td>
<td>607</td>
<td>21</td>
<td>18</td>
</tr>
<tr>
<td>PSOM</td>
<td>1</td>
<td>1000</td>
<td>1</td>
<td>15.9</td>
<td>972</td>
<td>676</td>
<td>23</td>
<td>19</td>
</tr>
</tbody>
</table>

Based on Tables 1 and 2, and Figures 7 to 9, the following observations were obtained:

- Generally, increasing the number of diodes in the model should improve the model performance. However, it increases the complexity of the system, the number of unknown parameters, and the computational time. Moreover, the results showed that the simulation time increased with increasing the number of diodes. On the other hand, SDM gave the best performance among the other models. The QtM came in the second place, followed by DDM, TDM and then QdM. Although QtM was in the second place, its performance was questionable. Some models of QtM were from best 50 models, while others were from the worst 50 models. This made the SDM to be the best candidate for the given set of data.

- Increasing the number of groups improved the performance. Better results could be achieved by increasing the number of groups instead of increasing the size of populations. From Table 3, the best performance was obtained from populations with size of 75 and 100, and number of groups of 37 and 50, respectively. The group itself consisted of two candidates, one leader, and one follower. The simulation time increased when the number of groups increased. However, this increment could be neglected, as it is less than 2%. Increasing number of groups reduced the box and whiskers sizes, and reduced outlier solutions.

- Generally, increasing the number of populations in metaheuristic techniques improved the method’s performance. However, it increased the computational time. For MGSSA, this was not fully true. The simulation time was increased as expected. This occurred due to the calculations of the followers, which took place in a recursive manner. On the other hand, the performance improvement was not guaranteed. The results showed that a population of 75 gave better results than a population of 500 by 2% for QtM.

- For the PSO and its versions, PSOTAC gave the best performance, followed by PSOI, PSOM and then PSO. It is worth to mention that the best performance of PSO versions had a RMSE that was almost 7.4 times the one obtained from the selected MGSSA. Moreover, it needed around eight times the simulation time and around 1.5 number of iterations to conclude its results compared to the proposed method.

- Increasing the number of diodes, groups, and/or population reduced the number of iterations required to reach a RMSE of 0.01 (A), and a RMSE of 0.02 (A).

Figure 7. Convergence rate of SSA at (a) 25, and (b) 500 population
5. CONCLUSIONS
In this work, the performance of PVSC was investigated using multi-group salp swarm algorithm (MGSSA). The MGSSA was proposed to estimate SDM, DDM, TDM, QdDM and QtDM of a PV solar panel.
Several groups (chains) were developed and their performance was evaluated using an I-V curve benchmark. The results were compared to PSOTAC, PSOM and PSOI in terms of RMSE, MAE, CR to RMSE values of 0.02 (A) and 0.01 (A), and the simulation time. MGSSAs with highest number of leaders had the best results compared to other algorithms in terms of CR, RMSE, and MAE. MGSSAs were better than PSO versions in term of CR, RMSE and MAE. Moreover, using high group numbers resulted in better model estimate with lower number of populations. The RMSE and MAE values could be reduced by 50 % by increasing leaders’ number or groups’ number. Based on this study, it is recommended to add diodes to the solar cell model, which would result in faster computing time and hence better performance of PVSC.

REFERENCES


