# Application of Moth-Flame Optimizer and Ant Lion Optimizer to Solve Optimal Reactive Power Dispatch Problems

Rebecca Ng Shin Mei[1], Mohd Herwan Sulaiman[1], Hamdan Daniyal[1] and Zuriani Mustaffa[2]
[1]*Faculty of Electrical & Electronics Engineering, University Malaysia Pahang, 26600 Pekan, Pahang, Malaysia.*
[2]*Faculty of Computer System & Software Engineering, University Malaysia Pahang, 26300 Gambang, Pahang, Malaysia.*
*rebeccamei527@gmail.com*

***Abstract*—This paper presents the application of two nature-inspired meta-heuristic algorithms, namely moth-flame optimizer (MFO) and ant lion optimizer (ALO) in obtaining the optimal settings of control variables for solving optimal reactive power dispatch (ORPD) problems. MFO is developed by the inspiration of the natural navigation method of moths during night time while ALO is inspired by the natural foraging technique of antlions in hunting ants. These two algorithms are implemented in ORPD to determine the optimal value of generator buses voltage, transformers tap setting and reactive compensators sizing in order to minimize power loss in the transmission system. In this paper, IEEE 57-bus system is utilized to show the effectiveness of MFO and ALO. Their statistical results are compared against other metaheuristic algorithms. The results of this paper illustrate that MFO is able to achieve a lower power loss than ALO and other selected algorithms from literature.**

***Index Terms*—Ant Lion Optimizer; Loss Minimization; Moth-Flame Optimizer; Optimal Reactive Power Dispatch.**

## I. INTRODUCTION

Optimal reactive power dispatch (ORPD) is a complex and nonlinear problem in power system operation. It is classified as a sub-problem of optimal power flow (OPF). There are numbers of objective functions of ORPD problems, including minimization of power loss, voltage deviation and voltage stability index [1]. In this paper, the objective function used to solve ORPD problems is through power loss minimization in power system. The power loss minimization is done by finding the optimized results of the control variables while satisfying the operating constraints. These control variables including generator buses voltage, transformers tap setting and reactive compensators setting.

From the past till now, there are numerous techniques have been proposed by researchers in addressing the ORPD problems. The techniques proposed ranging from conventional methods to meta-heuristic methods as well as hybrid optimization methods. Recently, meta-heuristic methods gain an ever-increasing interest in solving ORPD problems. The meta-heuristic methods are basically divided into three main categories: swarm intelligence, computation evolutionary and physic-based. Most of the techniques under meta-heuristic algorithms are proposed and developed according to the natural inspiration. Lately, many nature-inspired meta-heuristic algorithms have been applied to solve ORPD problems. This included artificial bee colony (ABC) [2], honey bee mating optimization (HBMO) [3], grey wolf optimizer (GWO) [4], cuckoo search algorithm (CSA) [5], harmony search algorithm (HSA) [1], gravitational search algorithm (GSA) [6], particle swarm optimization (PSO) [7]-[14] and so on.

This paper proposes two nature-inspired metaheuristic algorithms, moth-flame optimizer (MFO) and ant lion optimizer (ALO) in obtaining the optimal results of ORPD problem for power loss minimization objective. The optimization processes of MFO and ALO are independent of each other. The implementation of MFO in ORPD problems is through the concepts of natural navigation techniques of moth around a flame whereas ALO applied the concepts of natural foraging mechanism of antlion to solve ORPD problems. Both of these two algorithms have been developed by Seyedali Mirjalili [15], [16] in the year of 2015. The efficacy and effectiveness of MFO and ALO are tested by utilizing IEEE 57-bus system.

The organization of this paper is as follows: Section 2 discusses the ORPD mathematical formulation for power loss minimization objective. Then, Section 3 presents the brief introduction of MFO followed by brief description of ALO in Section 4. The implementation of MFO and ALO in solving ORPD problems is explained in Section 5. Section 6 analyses the simulation results along with the discussion. Last but not least, Section 7 concludes the findings of the study.

## II. ORPD MATHEMATICAL FORMULATION FOR LOSS MINIMIZATION

In this paper, the objective function of ORPD is to minimize total power loss of the transmission system. The ORPD problem can be formulated as the minimization of function $f(x,u)$ subjected to the expression below:

$$\begin{aligned} g(x,u) &= 0 \\ h(x,u) &\leq 0 \end{aligned} \tag{1}$$

where:

| | |
|---|---|
| $f(x,u)$ | = Objective function |
| $g(x,u)=0$ | = Equality constraints |
| $h(x,u)\leq 0$ | = Inequality constraints |
| $x$ | = Vector of dependent variables |
| $u$ | = Vector of control variables |

The function $f$ is subjected to the following operating constraints. The equality constraint is the power balanced of load flows which can be expressed as in Equation (2) and (3):

$$P_{Gi} - P_{Di} = V_i \sum_{j \in N_i} V_j \left( G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) \tag{2}$$

$$Q_{Gi} - Q_{Di} = V_i \sum_{j \in N_i} V_j \left( B_{ij} \cos\theta_{ij} - G_{ij} \sin\theta_{ij} \right) \tag{3}$$

where:
- $P_{Gi}$ = Real power generation
- $Q_{Gi}$ = Reactive power generation
- $P_{Di}$ = Real load demand
- $Q_{Di}$ = Reactive load demand
- $V_i$ = Voltage magnitude at *i*-th bus
- $V_j$ = Voltage magnitude at *j*-th bus
- $B_{ij}$ = Conductance of *i-j* th transmission line
- $G_{ij}$ = Susceptance of *i-j* th transmission line
- $\theta_{ij}$ = Angle difference between bus-*i* and bus-*j*

The inequality constraints including generators' constraints, transformers tap ratio and reactive compensators sizing are expressed in terms of their respective boundaries as below:

$$P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max} \quad i = 1, \ldots, N_G \tag{4}$$

$$Q_{Gi}^{\min} \le Q_{Gi} \le Q_{Gi}^{\max} \quad i = 1, \ldots, N_G \tag{5}$$

$$V_{Gi}^{\min} \le V_{Gi} \le V_{Gi}^{\max} \quad i = 1, \ldots, N_G \tag{6}$$

$$T_i^{\min} \le T_i \le T_i^{\max} \quad i = 1, \ldots, N_T \tag{7}$$

$$Q_{Ci}^{\min} \le Q_{Ci} \le Q_{Ci}^{\max} \quad i = 1, \ldots, N_C \tag{8}$$

where:
- $N_G$ = Number of generators
- $N_T$ = Number of transformers
- $N_C$ = Number of reactive compensators

In this paper, MATPOWER 5.1 software package [17], [18] is applied to achieve the objective function aforementioned. This software package is used to make sure fair and reasonable comparison can be made between the proposed algorithms with the selected reviewed techniques. Additionally, precise results can be obtained by executing the load flow program using MATPOWER.

## III. MOTH-FLAME OPTIMIZER (MFO)

MFO algorithm is inspired by the unique navigation techniques of moths during night time. They travel depending on the moonlight by using transverse orientation. In order to model MFO algorithm, the following matrices are expressed to represent the set of moths and flames, respectively:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & m_{n,d} \end{bmatrix} \tag{9}$$

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & F_{n,d} \end{bmatrix} \tag{10}$$

where:
- $n$ = Number of moths
- $d$ = Number of variables

In MFO, both moths and flames are solutions where moths are the actual search agents that navigate around the search space. On the other hand, the flames are the best position of moths obtained so far during optimization. The following mathematical formula expressed the mechanism of each moth updates its position according to a flame in order to find a better result [15]:

$$M_i = S\left(M_i, F_j\right) \tag{11}$$

where:
- $M_i$ = The *i*-th moth
- $F_j$ = The *j*-th flame

*S* is the logarithm spiral function which is the main update mechanism of moths as expressed as below:

$$S\left(M_i, F_j\right) = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \tag{12}$$

where:
- $b$ = Constant that used to define the shape of the logarithmic spiral
- $t$ = Random number that indicates how close the next position of moth to the flame
- $D_i$ = Distance of *i*-th moth for *j*-th flame

## IV. ANT LION OPTIMIZER (ALO)

ALO algorithm is another nature-inspired algorithm which is inspired by the natural foraging behaviour of antlions when hunting ants. It is developed according to five stages: random walk of ants, entrapment of ants, building pits, catching ants and rebuilding pits. In ALO, the ants' random walk positions are utilized and saved in matrix form as below:

$$M_{Ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix} \tag{13}$$

where:
- $n$ = Number of ants
- $d$ = Number of variables

The positions of antlions which hiding in traps somewhere in the search space also saved in matrix form as in Equation (14):

$$M_{Antlion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \cdots & AL_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ AL_{n,1} & AL_{n,2} & \cdots & AL_{n,d} \end{bmatrix} \tag{14}$$

where: $n$ = Number of antlions
$d$ = Number of variables

The entrapment of ants in antlions' traps can be mathematically expressed as below [16]:

$$c_i^t = Antlion_j^t + c^t \tag{15}$$

$$d_i^t = Antlion_j^t + d^t \tag{16}$$

where: $Antlion_j^t$ = Position of the selected $j$-th antlion at $t$-th iteration
$c^t$ = Minimum of all variables at $t$-th iteration
$d^t$ = Maximum of all variables at $t$-th iteration
$c_i^t$ = Minimum of $i$-th variable at $t$-th iteration
$d_i^t$ = Maximum of $i$-th variable at $t$-th iteration

Once an ant is in the trap, the antlions will try to slide the ants against towards them by shooting the sand outwards the center of the trap. This behavior can be described by the mathematic formulas below [16]:

$$c^t = \frac{c^t}{I} \tag{17}$$

$$d^t = \frac{d^t}{I} \tag{18}$$

where: $I$ = Ratio

Finally, the ant will become fitter than the antlion. This happened when the ant is caught by the anlion deeply in the trap. The antlion will then update its position according to the position of the hunted ant. This is to improve the chance for the next hunt. This situation can be expressed by the equation below [16]:

$$Antlion_j^t = Ant_i^t \quad if \quad f\left(Ant_i^t\right) > \left(Antlion_j^t\right) \tag{19}$$

where: $Antlion_j^t$ = Position of the selected $j$-th antlion at $t$-th iteration
$Ant_i^t$ = Position of the selected $i$-th ant at $t$-th iteration

The fittest antlion attained so far in each iteration is assumed as elite, which it is able to affect the random movement of the ants. Therefore, all the ants randomly move around the elite and a selected antlion simultaneously as in Equation (20) [16]:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{20}$$

where: $R_A^t$ = Random walk around the selected antlion at $t$-th iteration
$R_E^t$ = Random walk around the elite at $t$-th iteration

## V. MFO and ALO for ORPD Problem

The application of MFO and ALO in solving ORPD problems especially in finding the optimal setting of the control variables in order to achieve the power loss minimization by satisfying all the constraints aforementioned. It is worth to emphasize that the simulation processes of MFO and ALO are separate and independent. Initially, the number of search agents (number of moths and number of ants) and maximum iteration are set. Both of the moths and ants are the candidate solutions which constructed in matrix form as in Equation (9) and Equation (13), respectively.

During the evaluation process, each moth and each ant that comprises the base value of the control variables is mapped into the load flow data of MATPOWER. Then, the load flow program is executed to calculate the total power transmission loss. It is worth to mention that the processes of updating the positions (variables) using MFO and ALO are different. In MFO, the loss will be obtained for respected moth after updating the variables according to their corresponding flame using Equations (11)-(12). Whereas, in ALO, the loss will be obtained for respected antlion after updating the positions based on the ants using Equations (15)-(20). Then, the fittest antlion will be assumed as the elite.

Once the loss has been obtained, the matrix will be sorted according to their fitness value. The best result obtained so far is located at the top of the matrix while the worst result is situated at the bottom of the matrix. If the updated positions (variables) are out of the boundaries as constrained, they will be pegged at their respective lower and upper limits so as to ensure the results obtained are precise. The optimization will continue until the stopping criterion (maximum iteration) is reached. The application of MFO and ALO in solving ORPD is illustrated in Figure 1.

## VI. Results and Discussion

In order to illustrate the effectiveness of MFO and ALO algorithms in solving ORPD problems, a medium test system of IEEE-57 bus system is used in this paper. This test system consists of 25 control variables that need to be optimized which including seven generators, 15 transformers and three injected shunt reactive elements. The three reactive compensators are located at buses 18, 25 and 53, respectively. The operating boundaries of all control variables are tabulated in Table 1. For this case study, the real and reactive load demands are 1250.8 MW and 336.4 MVar, respectively. For optimization purpose, the number of search agents and maximum iteration is set as 30 and 300, respectively. The number of function evaluation (NFE) for this test case in order to reach the optimal results is 9000.

In this paper, the results of MFO and ALO are compared with four other nature-inspired meta-heuristic algorithms: firefly algorithm (FA) [19], grey wolf optimizer (GWO) [19], seeker optimization algorithm (SOA) [20] and cuckoo search algorithm (CSA) [5]. For fair and reasonable comparison, all the results of the selected reviewed algorithms are taken out and mapped into the same load flow program that used in this study. Their results of the optimized control variables are executed in order to calculate the total power transmission losses using MATPOWER. Table 2 tabulated the optimized results of the control variables and power losses obtained by different algorithms. The initial setting of the control variables of this test case also included in this table with base case loss of 27.8640 MW.

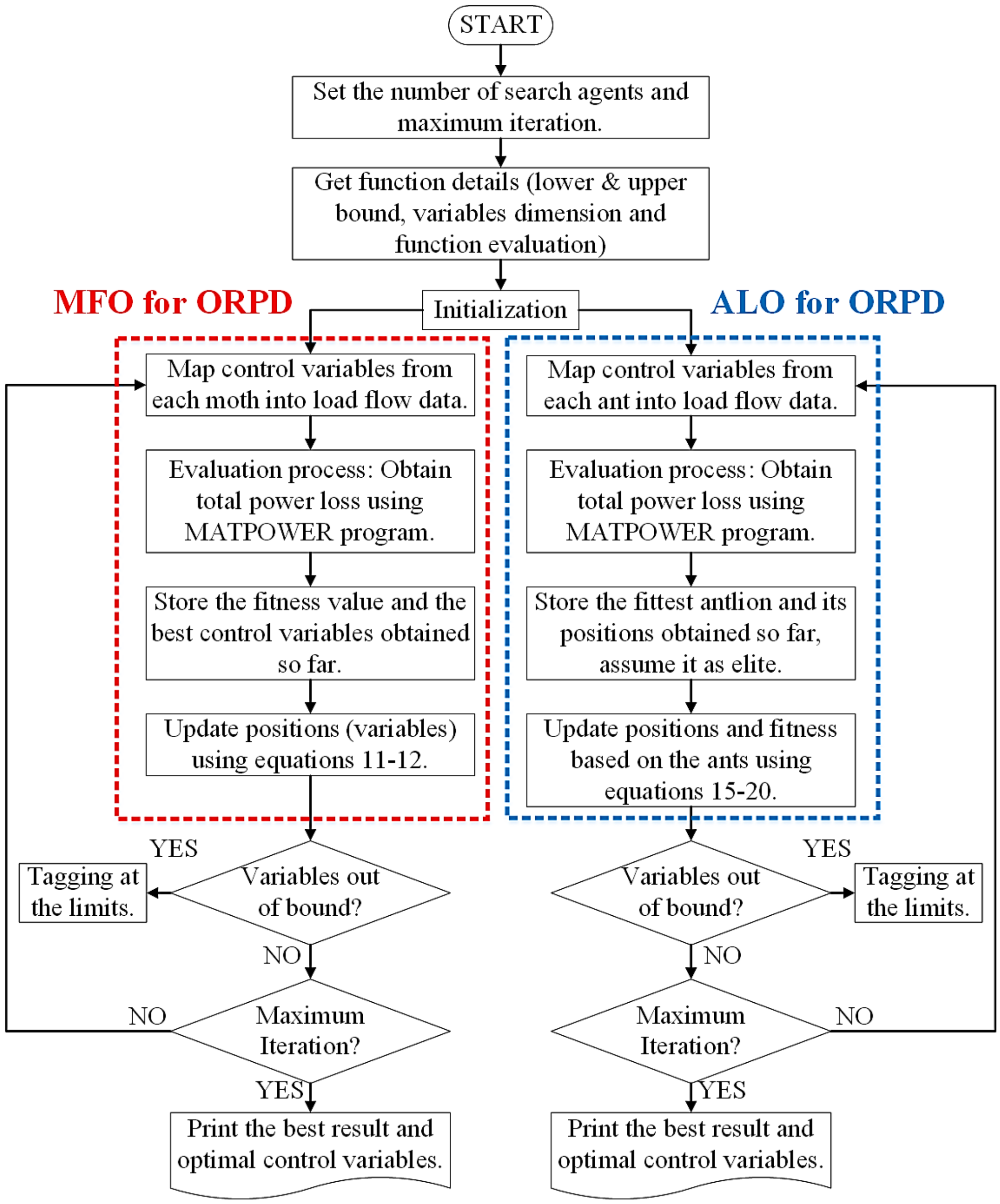


Figure 1: Flowchart of MFO and ALO for solving ORPD

Based on Table 2, it can be concluded that the power loss obtained by MFO is the best among others. Whereas, ALO get the worst result among all the algorithms tested in this case study. MFO is able to reduce 12.96 % of total power loss while ALO reduces 11.13 % of loss reduction from the base case loss. Furthermore, the recent best results attained from other study are those optimized by CSA ($P_{Loss}$=24.2619 MW) and SOA ($P_{Loss}$=24.2677 MW). When compared MFO with CSA and SOA, it produces about 0.06 % and 0.04 % of improvement in loss reduction. In a nutshell, it is concluded that MFO is able to excel their results. However, ALO produces a higher total power loss ($P_{Loss}$=24.7621 MW) than both CSA and SOA.

Table 3 illustrates the comparison of statistical results for power loss minimization between ALO and MFO in terms of best, average and worst results. Based on this table, MFO is able to gain lower best and average results than the results of ALO. Whereas, ALO is able to get a lower worst result than MFO. To further exhibit the comparison between ALO and MFO, their best-optimized results obtained from 30 simulation runs are plotted in the same graph as depicted in Figure 2. The results of power loss optimized by MFO are mostly varied between 24 MW and 25 MW while the results of ALO are mostly varied between 25 MW and 26 MW. From this graph, it can be concluded that MFO can produce a lower range of power losses than ALO. However, ALO can produce more consistent results than MFO throughout the 30 simulations. Furthermore, Figure 3 and Figure 4 show the convergence performances of MFO and ALO for power loss minimization in terms of power loss (MW) versus 300 iterations.

Table 1
Boundaries Setting of Control Variables for IEEE-57 Bus System

| Control Variables | Lower Bound | Upper Bound |
|---|---|---|
| Generator Buses Voltage | 0.94 p.u | 1.06 p.u |
| Transformers Tap Setting | 0.90 p.u | 1.10 p.u |
| $Q_{C18}$ | 0 MVar | 10.00 MVar |
| $Q_{C25}$ | 0 MVar | 5.90 MVar |
| $Q_{C53}$ | 0 MVar | 6.30 MVar |

Table 2
Results of Optimized Control Variables and Power Loss for IEEE-57 Bus System

| Control Variables | Initial (Base Case) | FA [19] | GWO [19] | SOA [20] | CSA [5] | ALO | MFO |
|---|---|---|---|---|---|---|---|
| $V_1$ | 1.0400 | 1.0600 | 1.0600 | 1.0600 | 1.0600 | 1.0600 | 1.0600 |
| $V_2$ | 1.0100 | 1.0572 | 1.0562 | 1.0580 | 1.0582 | 1.0595 | 1.0587 |
| $V_3$ | 0.9850 | 1.0428 | 1.0370 | 1.0437 | 1.0466 | 1.0494 | 1.0469 |
| $V_6$ | 0.9800 | 1.0366 | 1.0202 | 1.0352 | 1.0409 | 1.0409 | 1.0421 |
| $V_8$ | 1.0050 | 1.0541 | 1.0449 | 1.0548 | 1.0587 | 1.0600 | 1.0600 |
| $V_9$ | 0.9800 | 1.0355 | 1.0294 | 1.0369 | 1.0417 | 1.0469 | 1.0423 |
| $V_{12}$ | 1.0150 | 1.0320 | 1.0319 | 1.0336 | 1.0377 | 1.0426 | 1.0373 |
| $T_{4–18}$ | 0.9700 | 0.9312 | 0.9847 | 1.0000 | 0.9440 | 1.0791 | 0.9501 |
| $T_{4–18}$ | 0.9780 | 0.9901 | 0.9326 | 0.9600 | 1.0182 | 1.0629 | 1.0076 |
| $T_{21–20}$ | 1.0430 | 0.9845 | 0.9576 | 1.0100 | 1.0207 | 1.0471 | 1.0063 |
| $T_{24–26}$ | 1.0430 | 1.0112 | 0.9968 | 1.0100 | 1.0110 | 0.9993 | 1.0076 |
| $T_{7–29}$ | 0.9670 | 0.9683 | 0.9636 | 0.9700 | 0.9744 | 0.9768 | 0.9752 |
| $T_{34–32}$ | 0.9750 | 0.9657 | 0.9812 | 0.9700 | 0.9721 | 0.9985 | 0.9722 |
| $T_{11–41}$ | 0.9550 | 0.9762 | 1.0621 | 0.9000 | 0.9015 | 0.9958 | 0.9000 |
| $T_{15–45}$ | 0.9550 | 0.9653 | 0.9755 | 0.9700 | 0.9723 | 0.9827 | 0.9719 |
| $T_{14–46}$ | 0.9000 | 0.9524 | 0.9639 | 0.9500 | 0.9537 | 0.9793 | 0.9536 |
| $T_{10–51}$ | 0.9300 | 0.9671 | 0.9723 | 0.9600 | 0.9664 | 1.0204 | 0.9674 |
| $T_{13–49}$ | 0.8950 | 0.9291 | 0.9248 | 0.9200 | 0.9269 | 0.9530 | 0.9279 |
| $T_{11–43}$ | 0.9580 | 1.0020 | 0.9554 | 0.9600 | 0.9645 | 1.0092 | 0.9641 |
| $T_{40–56}$ | 0.9580 | 1.0224 | 1.1000 | 1.0000 | 0.9943 | 1.0675 | 0.9998 |
| $T_{39–57}$ | 0.9800 | 1.0232 | 0.9976 | 0.9600 | 0.9737 | 1.0480 | 0.9606 |
| $T_{9–55}$ | 0.9400 | 0.9687 | 0.9845 | 0.9700 | 0.9750 | 1.0111 | 0.9790 |
| $Q_{C18}$ | 10.000 | 4.1934 | 1.8917 | 9.9840 | 9.2807 | 8.8172 | 9.9968 |
| $Q_{C25}$ | 5.9000 | 4.2297 | 5.2489 | 5.9040 | 5.8943 | 5.3446 | 5.9000 |
| $Q_{C53}$ | 6.3000 | 5.9252 | 5.1513 | 6.2880 | 6.2885 | 5.4923 | 6.3000 |
| $P_{Loss}$ (MW) | 27.8640 | 24.4587 | 24.7523 | 24.2677 | 24.2619 | 24.7621 | 24.2529 |

Table 3
Comparison of Statistical Results for Power Losses Between ALO and MFO

| Compared Items ($P_{Loss}$) | ALO | MFO |
|---|---|---|
| Best Result (MW) | 24.7621 | 24.2530 |
| Average Result (MW) | 25.3026 | 24.7702 |
| Worst Result (MW) | 26.0480 | 26.3100 |

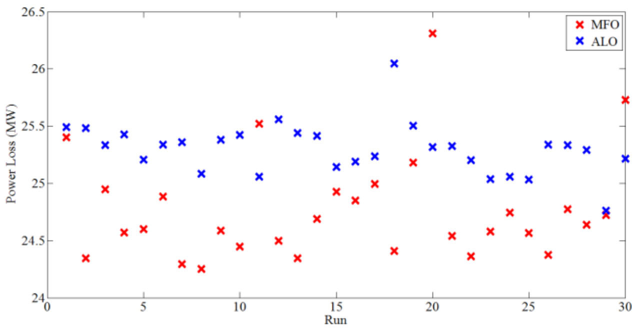


Figure 2: Comparison of power loss performances between ALO and MFO for 30 trail runs

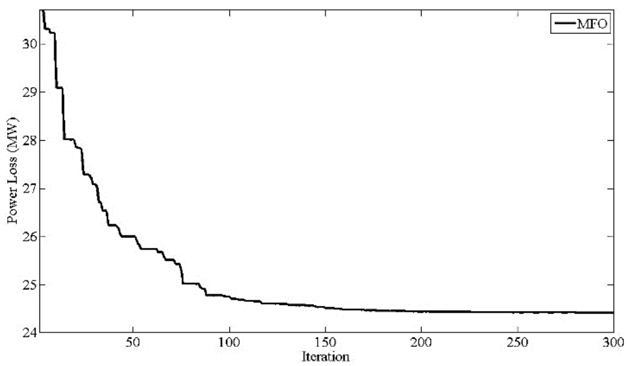


Figure 3: Convergence performance of MFO for power loss minimization

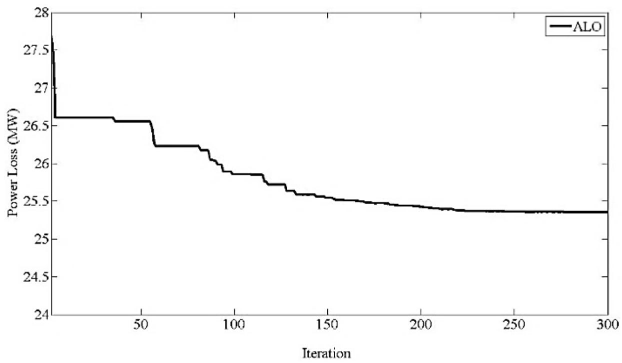


Figure 4: Convergence performance of ALO for power loss minimization

## VII. Conclusion

In this paper, two nature-inspired meta-heuristic algorithms, MFO and ALO are implemented in solving ORPD problems. The effectiveness of this two algorithms was tested utilizing IEEE 57-bus system. Based on the simulation results, it is proven that MFO is better compared to ALO and other reviewed algorithms from literature in terms of obtaining the lowest power loss. Whereas, ALO is the worst among the compared algorithms. However, ALO can produce more consistent results throughout the 30 simulations than MFO. Therefore, the implementation of this two algorithms in other applications including voltage deviation minimization, voltage stability index minimization, multi-objectives ORPD and considering practical operating constraints related to generating units (prohibited zones and valve points loading effects) are recommended to be proposed in future.

## ACKNOWLEDGMENT

This research study is supported by University Malaysia Pahang under UMP Post Graduate Research Grant Scheme (PGRS) #160378 and (RDU) #150362.

## REFERENCES

[1] A. H. Khazali and M. Kalantar, "Optimal reactive power dispatch based on harmony search algorithm," *Int. J. Electr. Power Energy Syst.*, vol. 33, no. 3, pp. 684–692, Jan. 2011.

[2] K. Ayan and U. Kılıç, "Artificial bee colony algorithm solution for optimal reactive power flow," *Appl. Soft Comput.*, vol. 12, no. 5, pp. 1477–1482, Jan. 2012.

[3] M. Ghasemi, S. Ghavidel, M. M. Ghanbarian, and A. Habibi, "A new hybrid algorithm for optimal reactive power dispatch problem with discrete and continuous control variables," *Appl. Soft Comput.*, vol. 22, pp. 126–140, May 2014.

[4] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, "Using the gray wolf optimizer for solving optimal reactive power dispatch problem," *Appl. Soft Comput.*, vol. 32, pp. 286–292, Apr. 2015.

[5] M. H. Sulaiman, M. I. Mohd Rashid, O. Aliman, M. R. Mohamed, A. Z. Ahmad, and M. S. Bakar, "Loss minimisation by optimal reactive power dispatch using cuckoo search algorithm," in *Clean Energy and Technology (CEAT) 2014, 3rd IET International Conference*, 2014, pp. 1–4.

[6] B. Shaw, V. Mukherjee, and S. P. Ghoshal, "Solution of reactive power dispatch of power systems by an opposition-based gravitational search algorithm," *Int. J. Electr. Power Energy Syst.*, vol. 55, pp. 29–40, Aug. 2014.

[7] R. P. Singh, V. Mukherjee, and S. P. Ghoshal, "Optimal reactive power dispatch by particle swarm optimization with an aging leader and challengers," *Appl. Soft Comput.*, vol. 29, pp. 298–309, Jan. 2015.

[8] P. Jangir, S. A. Parmar, I. N. Trivedi, and R. H. Bhesdadiya, "A novel hybrid particle swarm optimizer with multi verse optimizer for global numerical optimization and optimal reactive power dispatch problem," *Eng. Sci. Technol. an Int. J.*, vol. 20, pp. 570–586, Nov. 2016.

[9] E. Naderi, H. Narimani, M. Fathi, and M. R. Narimani, "A novel fuzzy adaptive configuration of particle swarm optimization to solve large-scale optimal reactive power dispatch," *Appl. Soft Comput.*, vol. 53, pp. 441–456, Jan. 2017.

[10] B. Zhao, C. X. Guo, and Y. J. Cao, "A multiagent-based particle swarm optimization approach for optimal reactive power dispatch," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 1070–1078, May 2005.

[11] K. Mahadevan and P. S. Kannan, "Comprehensive learning particle swarm optimization for reactive power dispatch," *Appl. Soft Comput.*, vol. 10, no. 2, pp. 641–652, Sept. 2009.

[12] M. Mehdinejad, B. Mohammadi-Ivatloo, R. Dadashzadeh-Bonab, and K. Zare, "Solution of optimal reactive power dispatch of power systems using hybrid particle swarm optimization and imperialist competitive algorithms," *Int. J. Electr. Power Energy Syst.*, vol. 83, pp. 104–116, Apr. 2016.

[13] H. Yoshida, K. Kawata, Y. Fukuyama, S. Takayama, and Y. Nakanishi, "A particle swarm optimization for reactive power and voltage control considering voltage security assessment," *IEEE Trans. power Syst.*, vol. 15, no. 4, pp. 1232–1239, 2000.

[14] Y. Li, Y. Cao, Z. Liu, Y. Liu, and Q. Jiang, "Dynamic optimal reactive power dispatch based on parallel particle swarm optimization algorithm," *Comput. Math. with Appl.*, vol. 57, no. 11, pp. 1835–1842, 2009.

[15] S. Mirjalili, "Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm," *Knowledge-Based Syst.*, vol. 89, pp. 228–249, Jul. 2015.

[16] S. Mirjalili, "The Ant Lion Optimizer," *Adv. Eng. Softw.*, vol. 83, pp. 80–98, Mar. 2015.

[17] R. D. Zimmerman, C. E. Murillo-Sanchez, and R. J. Thomas, "MATPOWER: Steady-state operations, planning, and analysis tools for power systems research and education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12–19, Feb. 2011.

[18] R. D. Zimmerman and C. E. Murillo-Sanchez, "Matpower 5.1 User's Manual." March, 2015.

[19] M. H. Sulaiman, Z. Mustaffa, H. Daniyal, M. R. Mohamed, and O. Aliman, "Solving optimal reactive power planning problem utilizing nature inspired computing techniques," *ARPN J. of Eng. and Appl. Sci.*, vol. 10, no. 21, Nov. 2015.

[20] C. Dai, W. Chen, Y. Zhu, and X. Zhang, "Seeker optimization algorithm for optimal reactive power dispatch," *IEEE Trans. Power Syst.*, vol. 24, no. 3, pp. 1218–1231, Aug. 2009.