STUDY OF NATURE INSPIRED COMPUTING (NIC) TECHNIQUE FOR OPTIMAL REACTIVE POWER DISPATCH PROBLEMS

MOHD HERWAN SULAIMAN

ZURIANI MUSTAFFA

OMAR ALIMAN

MOHD RUSLLIM MOHAMED

REBECCA NG SHIN MEI

RESEARCH VOT NO:

RDU150362

FAKULTI KEJURUTERAAN ELEKTRIK & ELEKTRONIK

UNIVERSITI MALAYSIA PAHANG

2017

Universiti Malaysia PAHANG
Engineering • Technology • Creativity

PUSAT PENGURUSAN PENYELIDIKAN (RMC)

BORANG PENGESAHAN LAPORAN AKHIR PENYELIDIKAN

TAJUK PROJEK : STUDY OF NATURE INSPIRED COMPUTING (NIC) TECHNIQUE FOR OPTIMAL REACTIVE POWER DISPATCH PROBLEMS

Saya : PROF MADYA DR. MOHD HERWAN SULAIMAN
(HURUF BESAR)

Mengaku membenarkan **Laporan Akhir Penyelidikan** ini disimpan di Perpustakaan Universiti Malaysia Pahang dengan syarat-syarat kegunaan seperti berikut :

1. Laporan Akhir Penyelidikan ini adalah hakmilik Universiti Malaysia Pahang
2. Perpustakaan Universiti Malaysia Pahang dibenarkan membuat salinan untuk tujuan rujukan sahaja.
3. Perpustakaan dibenarkan membuat penjualan salinan Laporan Akhir Penyelidikan ini bagi kategori TIDAK TERHAD.
4. * Sila tandakan (/)

[ ]	SULIT	(Mengandungi maklumat yang berdarjah keselamatan atau Kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972).
[ ]	TERHAD	(Mengandungi maklumat TERHAD yang telah ditentukan oleh Organisasi/badan di mana penyelidikan dijalankan).
[/]	TIDAK TERHAD	

Tandatangan & Cop Ketua Penyelidik

DR. MOHD HERWAN BIN SULAIMAN
ASSOCIATE PROFESSOR
FACULTY OF ELECTRICAL & ELECTRONICS ENGINEERING
UNIVERSITI MALAYSIA PAHANG
26600 PEKAN
PAHANG DARUL MAKMUR
TEL : 09-424 6140 FAX : 09-424 6111

Tarikh : 12 OKTOBER 2017

CATATAN : **Jika Laporan Akhir Penyelidikan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh laporan ini perlu dikelaskan sebagai SULIT dan TERHAD.*

ACKNOWLEDGEMENTS

First of all, thanks to Allah SWT on His blessing to make this research successful. Million thankful and appreciation to all research mates, whose encouragement, guidance and support from the initial to the final level with patience and knowledge enabled this group to develop an understanding of the project. The invaluable guidance and advice in this project has been a source of inspiration through the course of this research.

Finally, we would also like to express our sincere grateful to all our friends and students of Faculty of Electrical & Electronics Engineering for their helping during the time in need and who had directly or indirectly contributed in one way or another towards to accomplish of this project. Last but not least, thank you to all that always supported us in any respect during the completion of the project.

ABSTRACT

In this research, new nature-inspired meta-heuristic optimization algorithms namely moth-flame optimizer (MFO) and Ant Lion Optimizer (ALO) were implemented to address the optimal reactive power dispatch (ORPD) problems. MFO is developed based upon natural navigation technique of moths. This algorithm mimics the navigation characteristics of moths in order to travel according to the fittest position. The moths travel based upon the visible light sources as guidance during night time. ALO on the other hand is inspired by the foraging mechanism of antlions in catching preys. ALO is developed based upon five main stages: random walk of ants, entrapment of ants, building traps, catching preys and rebuilding traps. This research presents the realization of MFO and ALO in solving ORPD problems which is to investigate the optimal setting of control variables including generators voltage, transformers tap ratio and reactive compensators sizing in order to minimize transmission power loss and voltage deviation. ORPD problem is a nonlinear optimization problem that involving both equality constraints and inequality constraints. The proposed algorithms are tested on five different case studies which are IEEE 30-bus system with 13 control variables, IEEE 30-bus system with 19 control variables, IEEE 30-bus system with 25 control variables, IEEE 57-bus system with 25 control variables and IEEE 118-bus system with 77 control variables. The results from each case study were compared with the best results of other optimization algorithms that reported in the recent literatures in order to test the effectiveness of proposed MFO. The statistical simulation results of this project proved that MFO is able to produce compromising solutions by yielding the lowest power loss and voltage deviation among other reviewed algorithms. It can reduce 19.01 % (IEEE 30-bus system with 13 control variables), 20.47 % (IEEE 30-bus system with 19 control variables), 50.76 % (IEEE 30-bus system with 25 control variables), 12.96 % (IEEE 57-bus system) and 12.37 % (IEEE 118-bus system) of power losses from base case losses of each test case.

***Key researchers**: Assoc. Prof. Dr. Mohd Herwan Sulaiman, Dr. Zuriani Mustaffa, Omar Aliman, Assoc. Prof. Dr. Mohd Rusllim Mohamed and Rebecca Ng Shin Mei*

E-mail : herwan@ump.edu.my

Tel. No. : *094246140*

Vote No. : *RDU150362*

TABLE OF CONTENT

DECLARATION		
TITLE PAGE		
ACKNOWLEDGEMENTS		**ii**
ABSTRACT		**iiiii**
TABLE OF CONTENT		**iiv**
LIST OF TABLES		**vii**
LIST OF FIGURES		**iix**
LIST OF SYMBOLS		**x**
LIST OF ABBREVIATIONS		**xii**
CHAPTER 1 INTRODUCTION		**1**
1.1	Background of Research	1
1.2	Motivation and Problem Statement	2
1.3	Objectives	4
1.4	Scope and Limitations	4
1.5	Report Organization	5
CHAPTER 2 LITERATURE REVIEW		**6**
2.1	Introduction	6
2.2	Optimal Power Flow (OPF)	6
	2.2.1 Optimal Reactive Power Dispatch (ORPD)	7
2.3	Conventional Methods	7
2.4	Meta-Heuristic Methods	9

2.4.1 Evolutionary Computation (EC) 10

2.4.2 Physic-Based 14

2.4.3 Swarm Intelligence (SI) 16

2.5 Hybrid Methods 19

2.6 Summary 20

CHAPTER 3 METHODOLOGY 25

3.1 Introduction 25

3.2 Mathematical Formulation of Optimal Reactive Power Dispatch (ORPD) 25

3.2.1 Objective Functions 25

3.2.2 Equality Constraints 26

3.2.3 Inequality Constraints 27

3.2.4 Penalty Function Method 28

3.3 MATPOWER 29

3.4 Moth-Flame Optimizer (MFO) 30

3.4.1 Mathematical Formulation of MFO 31

3.4.2 Implementing MFO in ORPD Problem 34

3.5 Ant Lion Optimizer (ALO) 37

3.5.1 Mathematical Formulation of ALO 38

3.5.2 Implementing ALO in ORPD Problem 42

CHAPTER 4 RESULTS AND DISCUSSION 45

4.1 Introduction 45

4.2 Test Case 1: IEEE 30-Bus System with 13 Control Variables 46

4.3 Test Case 2: IEEE 30-Bus System with 19 Control Variables 56

4.4 Test Case 3: IEEE 30-Bus System with 25 Control Variables 58

4.5 Test Case 4: IEEE 57-Bus System with 25 Control Variables 64

4.6 Test Case 5: IEEE 118-Bus System with 77 Control Variables 71

CHAPTER 5 CONCLUSION AND RECOMMENDATIONS 83

5.1 Conclusion 83

5.2 Contribution 84

5.3 Recommendation for Future Research 85

REFERENCES 86

APPENDIX B LIST OF PUBLICATION 977

LIST OF TABLES

Table 4.1	Description of different case studies	46
Table 4.2	Overview of different test system characteristics	46
Table 4.3	Setting of variables limit for test case 1, IEEE 30-bus system	47
Table 4.4	Simulation parameters for DE, GA, ALO and MFO (test case 1)	47
Table 4.5	Optimal results of the control variables after optimization for test case 1, IEEE 30-bus system	50
Table 4.6	Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 1, IEEE 30-bus system)	51
Table 4.7	Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 1, IEEE 30-bus system)	51
Table 4.8	The percentage of loss reduction improved by MFO#30 compared with other optimization algorithms (test case 1, IEEE 30-bus system)	52
Table 4.9	Comparison of performances between 20 search agents and 30 search agents	55
Table 4.10	Comparison of voltage deviation for different optimization algorithms (test case 1, IEEE 30-bus system)	56
Table 4.11	Setting of variables limit for test case 2, IEEE 30-bus system	58
Table 4.12	Comparison of the proposed MFO with the results obtained by DE	58
Table 4.13	The lower and upper limits of the control variables along with the initial settings for test case 3, IEEE 30-bus system	60
Table 4.14	Optimal results of the control variables after optimization for test case 3, IEEE 30-bus system	61
Table 4.15	Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 3, IEEE 30-bus system)	61
Table 4.16	Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 3, IEEE 30-bus system)	62
Table 4.17	The percentage of loss reduction improved by MFO compared with other optimization algorithms (test case 3, IEEE 30-bus system)	62
Table 4.18	Comparison of power loss between ALO and MFO (test case 3, IEEE 30-bus system)	63
Table 4.19	Setting of variables limits for test case 4, IEEE 57-bus system	65
Table 4.20	Optimal results of the control variables after optimization for test case 4, IEEE 57-bus system	66
Table 4.21	Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 4, IEEE 57-bus system)	67

Table 4.22	Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 4, IEEE 57-bus system)	68
Table 4.23	The percentage of loss reduction improved by MFO compared with other optimization algorithms (test case 4, IEEE 57-bus system)	68
Table 4.24	Comparison of power loss between ALO and MFO (test case 4, IEEE 57-bus system)	70
Table 4.25	Setting of variables limits for test case 5, IEEE 118-bus system	72
Table 4.26	Optimal results of the control variables after optimization for test case 5, IEEE 118-bus system	73
Table 4.27	Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 5, IEEE 118-bus system)	77
Table 4.28	Comparison of percentage of loss reduction and computational time by MFO and other optimization algorithms (test case 5, IEEE 118-bus system)	77
Table 4.29	Statistical comparison of results based with power loss minimization objective (test case 5, IEEE 118-bus system)	78
Table 4.30	Comparison of simulation results of voltage deviation minimization objective for test case 5, IEEE 118-bus system	80

LIST OF FIGURES

Figure 2.1	Literature map	20
Figure 3.1	Transverse orientation of moth	30
Figure 3.2	Moth’s spiral flying path around close light source	31
Figure 3.3	Flowchart of MFO algorithm for solving ORPD problem	36
Figure 3.4	Pseudocode of MFO algorithm	37
Figure 3.5	Cone-shaped pit and hunting mechanism of antlion	38
Figure 3.6	Random walk of ant in antlion’s trap	41
Figure 3.7	Flowchart of ALO algorithm for solving ORPD problem	44
Figure 4.1	Convergence performances of power loss of IEEE 30-bus system for different numbers of search agents using MFO algorithm	53
Figure 4.2	Performance characteristics of MFO using 20 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)	54
Figure 4.3	Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)	54
Figure 4.4	Comparison of performances between 20 and 30 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)	55
Figure 4.5	Comparison of voltage deviation performances between ALO and MFO for 30 free running simulations (test case 1, IEEE 30-bus system)	56
Figure 4.6	Performance characteristics of MFO for 30 free running simulations (test case 3, IEEE 30-bus system)	63
Figure 4.7	Comparison of power loss performances between ALO and MFO for 30 free running simulations (test case 3, IEEE 30-bus system)	64
Figure 4.8	Convergence performances of power loss of IEEE 57-bus system for different numbers of search agents using MFO algorithm	69
Figure 4.9	Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 4, IEEE 57-bus system)	70
Figure 4.10	Comparison of power loss performances between ALO and MFO for 30 free running simulations (test case 4, IEEE 57-bus system)	71
Figure 4.11	Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 5, IEEE 118-bus system)	78
Figure 4.12	Convergence performances of power loss of IEEE 118-bus system for different numbers of search agents using MFO algorithm	79
Figure 4.13	Convergence characteristic of MFO for voltage deviation minimization (test case 5, IEEE 118-bus system)	82
Figure 4.14	Performance characteristics of MFO for voltage deviation minimization using 30 search agents for 30 free running simulations (test case 5, IEEE 118-bus system)	82

LIST OF SYMBOLS

x	Dependent variables vector
u	Control variables vector
Nl	Number of transmission lines
g_k	Conductance of k^{th} line
V_i	Voltage magnitude at load bus-i
V_j	Voltage magnitude at load bus-j
θ_i	Line angle at the line i
θ_j	Line angle at the line j
Nd	Number of load busses
V_i^{sp}	Specified value at load bus-i
B_{ij}	Susceptance between bus-i and bus-j
G_{ij}	Conductance between bus-i and bus-j
P_{Gi}	Real power generation
P_{Di}	Real load demand
Q_{Gi}	Reactive power generation
Q_{Di}	Reactive load demand
V_{Gi}	Bus voltage generation
N_G	Number of generators
T_i	Transformers tap ratio
N_T	Number of transformers
Q_{Ci}	Reactive compensators sizing
N_C	Number of reactive compensators
N_V^{lim}	Set of number of buses violating voltage limits
N_Q^{lim}	Set of number of buses violating injected reactive power limits
d	Number of variables
n	Number of moths
M	Matrix of moth
F	Matrix of flame
M_i	The i-th moth
F_j	The j-th flame
S	Logarithmic spiral function
b	A constant for defining the shape of the logarithmic spiral
t	A random number within the interval of [-1,1]
D_i	Distance of the i-th moth for j-th flame
l	Current iteration
N	Maximum number of flames
T	Maximum number of iterations
t	Step of random walk (current iteration)
$cumsum$	Cumulative sum
$r(t)$	Stochastic function
$rand$	A random number within the interval of [0,1]
M_{Ant}	Matrix of the position of each ant
M_{OA}	Matrix of the fitness of each ant
$M_{Antlion}$	Matrix of the position of each antlion
M_{OAL}	Matrix of the fitness of each antlion
n	Number of ants/antlions
$A_{i,j}$	Value of j-th variable of i-th ant/antlion
X_i^t	Min-max normalization equation
a_i	Minimum of random walk of i-th variable
c_i	Minimum of all variables for i-th ant
d_i	Maximum of all variables for i-th ant

c_i^t	Minimum of i-th variables at t-th iteration
d_i^t	Maximum of i-th variables at t-th iteration
$Antlion_j^t$	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
I	Ratio
w	A constant for defining the accuracy level of exploitation
Ant_i^t	Position of the selected i-th ant at t-th iteration
Ant_j^t	Position of the selected j-th ant at t-th iteration
R_A^t	Random walk around the antlion selected by the roulette wheel at t-th iteration
R_E^t	Random walk around the elite at t-th iteration
F	Step size for DE
CR	Crossover rate for DE
Pc	Crossover probability for GA
P_m	Mutation probability for GA
P_G	Generator real power output

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
ACROA	Artificial Chemical Reaction Optimization Algorithm
AGA	Adaptive Genetic Algorithm
AKHA	Antarctic Krill Herd Algorithm
ALO	Ant Lion Optimizer
AS	Ant System
BA	Bat Algorithm
BBO	Biogeography-Based Optimization
BH	Black Hole
CFO	Central Force Optimization
CLPSO	Comprehensive Learning Particle Swarm Optimization
COA	Cuckoo Optimization Algorithm
CPVEIHBMO	Chaotic Parallel Vector Evaluated Interactive Honey Bee Mating Optimization
CRO	Chemical Reaction Optimization
CS	Cuckoo Search
CSS	Charged System Search
DE	Differential Evolution
DE	Dolphin Echolocation
DE-AS	Differential Evolution and Ant System
EAs	Evolutionary Algorithms
EC	Evolutionary Computation
EMA	Exchange Market Algorithm
EP	Evolutionary Programming
ES	Evolutionary Strategy
FA	Firefly Algorithm
FAHCLPSO	Fuzzy Adaptive Heterogeneous Comprehensive Learning Particle Swarm Optimization
FAPSO	Fuzzy Adaptive Particle Swarm Optimization
FFOA	Fruit Fly Optimization Algorithm
FPA	Flower Pollination Algorithm
GA	Genetic Algorithm
GA-IPM	Genetic Algorithm and Interior Point Method
GS	Genetic Search
GSA	Gravitational Search Algorithm
GWO	Grey Wolf Optimizer
HBMO	Honey Bee Mating Optimization
HSA	Harmony Search Algorithm
IBA	Improved Baboon Algorithm
ICA	Imperialist Competitive Algorithm
IGSA-CSS	Improved Gravitational Search Algorithm-based Conditional Selection Strategies
IHSA	Improved Harmony Search Algorithm
IPM	Interior Point Method
IWO	Invasive Weed Optimization
KGMO	Kinetic Gas Molecules Optimization
LP	Linear Programming
MAPSO	Multiagent-based Particle Swarm Optimization
MDEA	Modified Differential Evolution Algorithm
MFO	Moth-Flame Optimizer
MICA	Modified Imperialist Competitive Algorithm

MICA-IWO	Modified Imperialist Competitive Algorithm and Invasive Weed Optimization
MINLP	Mixed Integer Non-Linear Programming
MVO	Multi-Verse Optimizer
NFE	Number of Function Evaluation
NFL	No-Free-Lunch
NLP	Non-Linear Programming
OGSA	Opposition-based Gravitational Search Algorithm
OPF	Optimal Power Flow
ORPD	Optimal Reactive Power Dispatch
PSO	Particle Swarm Optimization
PSO-GSA	Particle Swarm Optimization and Gravitational Search Algorithm
PSO-ICA	Particle Swarm Optimization and Imperialist Competitive Algorithm
PSO-MVO	Particle Swarm Optimization and Multi-Verse Optimizer
QEA	Quantum-inspired Evolutionary Algorithm
QOCRO	Quasi-Oppositional Chemical Reaction Optimization
QP	Quadratic Programming
SARGA	Self-Adaptive Real coded Genetic Algorithm
SBX	Simulated Binary Crossover
SGA	Simple Genetic Algorithm
SI	Swarm Intelligence
SMS	Start of Mater Search
SOA	Seeker Optimization Algorithm
SQP	Sequential Quadratic Programming
TS	Tabu Search

CHAPTER 1

INTRODUCTION

1.1 Background of Research

Over the last few decades, electrical power system has become increasingly significant due to the fast development of world economy. The electrical power system acts as systems that generate, supply, transmit and distribute electric power for a variety of load demands. It is also the core of renewable energy system. Undeniably, once the demands for electric power increased, which greatly depend on both non-renewable energy and renewable energy, the consumption of these energy resources will also increase gradually. Additionally, unreasonable and uncontrollable distribution of electric power will also increase the total power transmission loss as well as the power quality of the system.

In the year of 1977, the accident of New York blackout has been proven is due to the cause of reactive power problem. Again, in year 1987, the blackout in Tokyo was also caused by reactive power shortage during peak load hours (Chebbo, Irving, & Sterling, 1992). Undeniably, both of these incidents cause social disruptions, economic losses and inconvenient to the public. Through these blackouts, it is proven that the vitality of reactive power planning, controlling and dispatching in retaining the system security, reduce total power transmission losses and decrease generation cost.

In latest development on power system research study, optimal reactive power dispatch (ORPD) has received an ever-increasing interest and attention as it plays a significant role in the operation of power system on both security aspects and economic issues. ORPD is a particular form of optimal power flow (OPF) calculation. It is defined as a well-known nonlinear and non-convex optimization problem in power system which involving both discrete and continuous control variables while satisfying various equality and inequality constraints (Ayan & Kılıç, 2012; Serhat Duman, Güvenç, Sönmez, & Yörükeren, 2012; A. Ghasemi, Valipour, & Tohidi, 2014; M. Ghasemi, Ghavidel, Ghanbarian, & Habibi, 2014; Khazali & Kalantar, 2011; Shaw, Mukherjee, & Ghoshal, 2014; Mohd Herwan Sulaiman, Mustaffa, Mohamed, & Aliman, 2015; Varadarajan & Swarup, 2008a, 2008b; Zhao, Guo, & Cao, 2005).

Undeniably, the prominent aim behind applying ORPD problem in enhancing the power system operation is to reallocate reactive power in the system in such a way that the minimum total transmission losses and enhancement the voltage profile can be achieved (M. Ghasemi et al., 2014). Therefore, in order to minimize the transmission losses and other objective functions, the optimal setting for all controllable variables need to be identified which including transformers tap ratio, generator buses voltage and reactive compensators sizing.

This research presents an implementation of recently proposed nature-inspired heuristic techniques, namely moth-flame optimization (MFO) algorithm and ant lion optimizer (ALO) in solving ORPD problem separately. Both MFO and ALO algorithms were developed by Seyedali Mirjalili in year 2015 (Mirjalili, 2015a, 2015b). In this research study, the main solution for solving ORPD problem is by using MFO algorithm while ALO is an alternative method used as a comparison. The main objective of ORPD is to minimize the total transmission line losses and voltage deviation respectively. The proposed optimization algorithms in this research will be implemented to find the optimal setting of control variables that minimizes the objective functions of ORPD problem. MFO algorithm was expected to contribute satisfaction optimal results in solving ORPD problems than ALO algorithm and other reviewed techniques.

1.2 Motivation and Problem Statement

In power system, the load demands keep changing continuously along with the development of the economy. Thence, the scales of the power grid keep improving in order to combat the growing demands. Nevertheless, the construction of the power grid in some places did not upgrade or improve with the growth of the load demands. This will consequently causes serve shortage in reactive power. Undeniably, reactive power is critical on the operation of power system in terms of economic and safety aspects.

Unreasonable distribution of reactive power can greatly influence the power quality; causes increased in transmission power loss and affect the voltage stability of the power system. Each year, a large amount of electricity is wasted as power loss on the distribution and transmission lines. This power loss not only will cause increase in generation cost, but also produces energy waste and carbon emission. Therefore, in order to minimize power loss and improve voltage stability, the simple, economical and practical technique is thru optimal reactive power dispatch (ORPD) method. The starting point of ORPD in the early days is by installing reactive compensators in order to improve the power factor.

ORPD is modelled as a large scale mixed integer nonlinear programming (MINLP) problem. This problem consist a mixture of discrete and continuous variables. The transformers tap ratios and outputs of shunt capacitors or reactors have discrete characteristic, while static VAR compensators, reactive power outputs of generators, bus voltage magnitudes and angles are classified as continuous variables (Ayan & Kılıç, 2012; Khazali & Kalantar, 2011; Zhao et al., 2005). ORPD problem is a popular topic in power system where there has been numbers of algorithms ranging from conventional methods to meta-heuristic methods and even hybrid methods being developed in solving this problem. However, conventional methods are inefficient in overcoming nonlinear, discrete functions and constraints, which this would lead to loss of accuracy when solving ORPD problem. On the other hand, meta-heuristic methods are able to handle the disadvantages faced by conventional methods (Shaw et al., 2014) and they are superior in obtaining global optimum.

Despite meta-heuristic methods can attain near global optimum solutions, there are still some evolutionary computation (EC) techniques which are ineffective in managing integer and discrete nature (Ayan & Kılıç, 2012; A. Ghasemi et al., 2014). This consequently will cause the solutions far from the global optimum. Furthermore, hybrid methods which are either the combination of conventional method and meta-heuristic method or combination between two meta-heuristic methods are able to provide better results with their global search ability and rapid convergence speed (M. Ghasemi et al., 2014; Huang & Huang, 2012). In year 2015, recently developed nature inspired computation technique namely, grey wolf optimizer (GWO) is able to achieve better results than the hybrid method in solving ORPD problems as reported in literature (Mohd Herwan Sulaiman et al., 2015).

Undeniably, all these optimization algorithms are distinct from their effectiveness, simplification as well as convergence speed. Thence, there is no exact proof that which optimization technique is the best for solving ORPD problem. As according to no-free-lunch (NFL) theorem (Wolpert & Macready, 1997), stated that there is no specific algorithm that is able to solve all the optimization problems. Therefore, ORPD problem still can be solved by newly developed optimization algorithms.

1.3 Objectives

The prominent purpose of this research is to implement moth-flame optimizer (MFO) and ant lion optimizer (ALO) for solving optimal reactive power dispatch (ORPD) problem. Then, the obtained optimal results of the proposed algorithms are compared with selected optimization techniques in literatures.

1.4 Scope and Limitations

This research study is mainly focused on solving the optimal reactive power dispatch (ORPD) problem. It should be emphasized that this research covers only the minimization of total power transmission losses and voltage deviation separately. On the other hand, the multi objective ORPD optimization is not part of the research study. Furthermore, for the optimization algorithms, two recently developed nature-inspired computation techniques which are moth-flame optimizer (MFO) and ant lion optimizer (ALO) will be implemented separately in the research. Both of these optimization algorithms are developed in year 2015 (Mirjalili, 2015a, 2015b). It should be emphasized that the main algorithm for this research is MFO instead of ALO. In this research, the results obtained from ALO are just used as a comparison for MFO.

There are three test systems applied in this research in order to test the effectiveness of the proposed algorithms in minimizing transmission losses and voltage deviation. The test systems included IEEE-30 bus system, IEEE-57 bus system and

IEEE-118 bus system. In addition, there are three different test cases for IEEE-30 bus system that implemented in this study. Firstly, IEEE-30 bus system with 13 control variables which considers load demands of 283.2 MW (real power) and 126.2 MVar (reactive power) to be tested. Secondly, IEEE-30 bus system with 19 control variables of the same load demands was tested. Thirdly, the test case of IEEE-30 bus system which considers 25 control variables of the same load demands was tested. Followed by IEEE-57 bus system with 25 control variables which considers load demands of 1250.8 MW and 336.4 MVar. And lastly, IEEE-118 bus system with 77 control variables by considering load demands of 4242 MW and 1438 MVar.

1.5 Report Organization

This report consists of five chapters. Chapter 1 explains the background of the research on optimal reactive power dispatch (ORPD) problem, the motivation and problem statement, followed by the objectives of this research study and lastly the scopes and limitations of this project. The organization of the report also included in Chapter 1.

Chapter 2 discusses and explains the study on literatures that related to this project. Different types of optimization techniques in solving ORPD problem, ranging from conventional methods to newly developed algorithms are presented.

Chapter 3 presented the mathematical formulation of ORPD problem, including formulation for transmission loss and voltage deviation. The introduction on moth-flame optimization (MFO) algorithm and ant lion optimizer (ALO) also presented, followed by the implementation of MFO and ALO in solving ORPD problem.

Chapter 4 analyses and discusses the simulation results obtained by implementing MFO algorithm in ORPD problem which tested on five case studies.

Last but not least, Chapter 5 presents the conclusion of this research.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter discusses the review on optimal reactive power dispatch (ORPD) problem in power system. Various types of optimization methods proposed by the researchers in overcoming ORPD problem also deeply discussed in this chapter.

2.2 Optimal Power Flow (OPF)

Optimal power flow (OPF) study or known as load flow study is an analysis method of the flow of electric power in the power system. In other word, it is an analysis on the capability of the system to adequately supply electricity to the connected load and fulfil customer demand. OPF study is essentially important in identifying the best operation of existing systems and planning future expansion of modern power systems. In general, OPF problem is defined as a large-scale, highly constrained, nonlinear and non-convex optimization problem (Serhat Duman et al., 2012; Varadarajan & Swarup, 2008a). It is an effective way to enhance the voltage profile, minimize network losses and keep the system running under normal conditions. The various objectives of OPF problem included minimization of generation cost, optimal reactive power dispatch (ORPD), shift in controls, VAR investment cost and maximization of social benefit (Varadarajan & Swarup, 2008a). In year 1968, the method of OPF was firstly proposed by Dommel and Tinney (Dommel & Tinney, 1968), which OPF find the optimum solution and balance the power flow equations by utilizing an optimization technique (Sinsuphan, Leeton, & Kulworawanichpong, 2013).

2.2.1 Optimal Reactive Power Dispatch (ORPD)

Optimal reactive power dispatch (ORPD) is a sub problem of optimal power flow (OPF) calculation. It is defined as a nonlinear optimization problem in power system which involving both continuous and discrete control variables while satisfying the physical and operating constraints (Mohd Herwan Sulaiman et al., 2015). The main objective of ORPD is to minimize power losses, improve voltage profile and voltage stability through reallocation of reactive power in the system (El Ela, Abido, & Spea, 2011). Up to now, ORPD problem has received an ever-increasing attention. It has been solved by a variety number of optimization techniques ranging from conventional methods to newly developed meta-heuristic methods.

2.3 Conventional Methods

Over the last few decades, a number of conventional optimization methods have been developed and implemented to solve the ORPD problem. In this subsection, the conventional optimization techniques that will be discussed include linear programming (LP), non-linear programming (NLP), quadratic programming (QP), gradient method, Newton method of solutions and interior point method (IPM).

Linear programming (LP) is one of the conventional method which has been utilized in solving ORPD problems (Alsac, Bright, Prais, & Stott, 1990; Kirschen & Van Meeteren, 1988). The main advantages of the LP are its robustness, reliability and solution speed in overcoming an optimization problem. These characteristics are very useful and helpful for application like security optimization. However, there are limitation in which both the objective function and constraints need to have linear relationship when using the LP technique to solve optimization problem (Varadarajan & Swarup, 2008a). This consequently may cause loss of accuracy and the solutions obtained may be far from the global optimum.

In (Pudjianto, Ahmed, & Strbac, 2002), D. Pudjianto et al. proposed linear programming (LP) and non-linear programming (NLP) direct reactive OPF. This method is used to distribute reactive power between competing generators in a deregulated situation. It was found that the total cost of the system which are associated with the concern of reactive power was convincingly precise by applying LP method. On the other hand, NLP offers precision for the solution and shorter convergence speed.

However, the convergence of the optimization problem using NLP could not be assured for every situation.

Apart from LP and NLP, quadratic programming (QP) is another method which has been considered by a number of researches in solving ORPD problem (Grudinin, 1998; Lin, David, & Yu, 2003; Momoh, Guo, Ogbuobiri, & Adapa, 1994; Perez Abril & Quintero, 2003). QP is advantaged over the LP method since it is more adaptable to the nonlinear behaviour of ORPD problem. Whereas, the shortcoming is that it is not efficient for high dimensional problems where the computation time will increase with the dimension. Furthermore, both QP and NLP serve limitations when dealing functions and constraints with nonlinear and discontinuous properties as well as function with numerous local minima (Khazali & Kalantar, 2011; Zhao et al., 2005).

Interior point method (IPM) or known as barrier method is another conventional technique that has been widely utilized in overcoming the OPF problem of large-scale power systems (Granville, 1994; Khazali & Kalantar, 2011; Momoh et al., 1994; Y.-C. Wu, Debs, & Marsten, 1994; Zhao et al., 2005). In (Granville, 1994), S. Granville overcame ORPD problem using IPM. This technique was designed based upon the primal dual logarithmic barrier method. It is concluded that IPM required less number of control variables and iteration number. The number of iterations is insensitive to the number of control variables or the size of the system. Moreover, this method also offers the following characteristics: robustness, no active set identification problems, efficiency in handling loss reduction matters and the optimal reactive allocation for ill-conditioned systems.

Furthermore, the advantages of IPM, which offer rapid convergence and convenience in dealing with inequality constraints (Ayan & Kılıç, 2012; Khazali & Kalantar, 2011; Zhao et al., 2005) help to encounter the weaknesses of gradient method and Newton method of solutions. Both of these two methods serve problems when handling inequality constraints (Varadarajan & Swarup, 2008a). Nevertheless, the IPM based method is ineffective in solving nonlinear, discrete functions and constraints as well as function having multiple local minima. Unfortunately, ORPD problem does have all these characteristics (Khazali & Kalantar, 2011; Zhao et al., 2005).

In spite of that conventional methods are computationally fast, however, they are inefficient in handling non-convex and nonlinear ORPD problem, which causes most of the conventional methods converge to local optima (El Ela et al., 2011). On the other hand, the traditional optimization techniques require heavy computation burden which causes them face difficulty in obtaining optimal results in the aspect of accuracy and rapidity (A. Ghasemi et al., 2014). Moreover, the optimization process of some traditional techniques including QP, NLP and IPM suffer from local minima entrapment which this may result in obtaining solutions that are non-optimal (El Ela et al., 2011; A. Ghasemi et al., 2014; Khazali & Kalantar, 2011; Zhao et al., 2005). In short, these methods lose their efficiency due to algorithmic complexity. Additionally, they are further restricted by their insecure convergence, robustness, sensitive to the initial search point and inefficient in dealing with discrete variables (Abido, 2002; Grudinin, 1998; Varadarajan & Swarup, 2008a).

2.4 Meta-Heuristic Methods

In recent years, the so-called stochastic optimization algorithms or known as meta-heuristic methods have been used as prime techniques for getting the optimal solutions of practical engineering problems (Blum, Puchinger, Raidl, & Roli, 2011; BoussaïD, Lepagnot, & Siarry, 2013; Gogna & Tayal, 2013). These methods are proposed to alleviate the aforementioned drawbacks of conventional optimization methods. Undeniably, traditional optimization methods are mostly deterministic that suffer from local optima stagnation (Mirjalili, 2015a). In other word, it is an entrapment of an optimization algorithm in local optimum solutions that consequently will result in failing to find the global optimum solutions (Mirjalili, 2015b). This disadvantage make them highly inefficient in overcoming practical optimization problems.

However, latter development in meta-heuristic techniques have given vast benefit in solving optimization problems by utilizing their stochastic operators (Bianchi, Dorigo, Gambardella, & Gutjahr, 2009). These stochastic operators make them advantage over conventional optimization methods. In addition, randomness is the main characteristic of meta-heuristic algorithms (Hoos & Stützle, 2004) which they employed random operators when searching for global optima in a search area. Although the characteristic results in obtaining different solutions to a given problem in each run, they are able to avoid local solution effectively than conventional approaches (Kirkpatrick, 1984).

As reported in the literature, meta-heuristic methods are able to produce better solutions in solving ORPD problems (Mohd Herwan Sulaiman et al., 2015). As a summary, meta-heuristic methods can be classed into three main categories based on the source of nature inspiration which are evolutionary computation (EC), physics-based and swarm intelligence (SI) (A. Ghasemi et al., 2014; Mirjalili, 2015a; Zhao et al., 2005). In this section, different meta-heuristic optimization algorithms will be discussed.

2.4.1 Evolutionary Computation (EC)

Evolutionary algorithms (EAs) are methods that mimic the evolutionary processes in nature (Mirjalili, 2015a). They seek for global optimum in a search region by producing one or more random solution for a given problem (Talbi, 2009). These random solutions are known as candidate solutions. Then, these sets of candidate solutions are improved iteratively until the satisfaction of a terminating condition. Thence, by running the evolutionary algorithm several times, the probability of achieving better results near global optimum will be increased, even though it is not guaranteed to obtain a very accurate approximation of global optimum. Furthermore, EAs have the following benefits including simplicity due to their natural evolutionary principles, problem and derivation independence as well as local optima avoidance (Mirjalili, 2015b).

In the past, evolutionary computation techniques such as evolutionary programming (EP) (Happ, 1977; Q Hw Wu & Ma, 1995; Yan, Lu, & Yu, 2004), hybrid EP (Yan et al., 2004), evolutionary strategy (ES) (Gomes & Saavedra, 2002), genetic search (GS) (Q H Wu & Ma, 1994), genetic algorithm (GA) (Durairaj, Devaraj, & Kannan, 2006; Iba, 1994; Q H Wu, Cao, & Wen, 1998), improved GA (Devaraj, 2007), real parameter GA (Devaraj, Durairaj, & Kannan, 2008), adaptive GA (Q H Wu et al., 1998), tabu search (TS) (Nualhong, Chusanapiputt, Phomvuttisarn, & Jantarang, 2004) and differential evolution (DE) (El Ela et al., 2011; Liang, Chung, Wong, Duan, & Tse, 2007; Varadarajan & Swarup, 2008c) have been utilized and successful in solving ORPD problem. The biggest advantages of EAs over conventional methods is that they do not need to simplify the objective functions as they are superior in dealing with non-convex and discontinuous problems. In other word, they do not require both of the objective functions and constraints to be continuous and differentiable (Lee & El-Sharkawi, 2008). Although EAs are highly efficient in obtaining the global best solutions, however, their drawback is that they are not efficient in fast convergence, which mean they are weak in obtaining the global optimum in the shortest period (Vlachogiannis & Lee, 2006).

Despite of the shortcoming of EAs, the interest of EAs in terms of their search power and modelling capability have promoted their ability in overcoming ORPD problems in power system (Iba, 1994; Lai & Ma, 1997; Lee, Bai, & Park, 1995; Ma & Lai, 1996; Q Hw Wu & Ma, 1995). Evolutionary algorithms including EP, DE, GA and self-adaptive real coded GA (SARGA) will further be discussed in this section.

2.4.1.1 Evolutionary Programming (EP)

Evolutionary programming (EP) is one of the optimization algorithm fall under the category of evolutionary computation techniques (Miranda, Srinivasan, & Proenca, 1998). Years back from (Lai & Ma, 1997; Ma & Lai, 1996; Q Hw Wu & Ma, 1995), the ORPD problem had been solved by a number of researchers utilizing EP technique. In (Lai & Ma, 1997), the paper reported that EP has a higher ability in solving non-continuous problems compared to nonlinear programming. From the reviewed literatures, EP is proven to be able to avoid suffering from local optimal entrapment. It is able to solve ORPD problem effectively by obtaining global or near global minimum solutions.

EP operates in a different method as compared to conventional optimization techniques where the objective functions and constraints of the ORPD problem does not need to be differentiated. The EP undergoes three processes: mutation, competition as well as reproduction. In EP, it elects individuals in a population to regenerate new generations by utilizing probability transition rules. The individuals within the old generation and mutated old generation compete against each other. Finally, the fittest individual form the next generation (Q Hw Wu & Ma, 1995).

2.4.1.2 Differential Evolutionary (DE)

In the year 1995, differential evolutionary (DE) was first introduced by Storn and Price (Storn & Price, 1995). DE is a population based stochastic algorithm for global optimization problems (Varadarajan & Swarup, 2008a). It has been utilized as the solution to ORPD problem by a number of researchers (El Ela et al., 2011; Varadarajan & Swarup, 2008a, 2008b). DE is proposed by the inspiration of sociological and biological motivations (El Ela et al., 2011). It has fast computational speed and parallel search characteristics (Huang & Huang, 2012). DE is advantaged over conventional methods by its three benefits: it is easy to use in terms of simplicity in coding and it can find near optimal results regardless the initial parameter values. Moreover, its computational time is fast and it only needs few numbers of control parameters (El Ela et al., 2011). Furthermore, DE is able to deal with optimization problems of discrete nature and integer (Das, Abraham, & Konar, 2008; KARABOĞA & Ökdem, 2004; Storn & Price, 1995, 1997) where the ORPD problem does has these characteristics.

DE is different from other EAs in two phases: mutation as well as recombination where it perturbs the population by utilizing weighted differences between solution vectors (Varadarajan & Swarup, 2008a). In (Varadarajan & Swarup, 2008a), DE algorithm based OPF is implemented for ORPD as well as voltage control in power system planning and operation. In the study, it is found that DE required fewer control parameters than well-known method, particle swarm optimization (PSO). These control parameters included population size, step size and crossover rate. Furthermore, the study made by Vesterstrom and Thomsen claimed that DE is better than PSO and EAs as it is able to obtain the lowest fitness solution for most of the problems reported in the literature (Vesterstrom & Thomsen, 2004). Again, in (KARABOĞA & Ökdem, 2004), it is reported that the convergence speed of DE is significantly faster than GA. Additionally, DE is robust as it is able to reproduce the same solutions consistently over many simulations as compared to PSO. However, PSO is more dependent on the randomized initialization of the individuals (Vesterstrom & Thomsen, 2004).

2.4.1.3 Genetic Algorithm (GA)

Genetic algorithm (GA) is another evolutionary computation technique which has been applied in solving ORPD problems (Devaraj & Roselyn, 2010; Iba, 1994). GA (Golberg, 1989) is a blind search algorithm utilizing stochastic operations. It is inspired by the mechanics of natural genetics. The operators of GA are simple, involving three processes: mutation, reproduction as well as crossover (Iba, 1994). Initially, starting with an initial population, GA utilizes the information or knowledge contained in the current population. Then, it explores new individuals by producing offspring using the aforementioned operators which can then take over members of the old generation.

Recently, a number of modifications have been made to the original GA so as to enhance the effectiveness of GA in solving ORPD problem. In (Bakirtzis, Biskas, Zoumas, & Petridis, 2002), Bakirtzis et al. overcoming optimal power flow problems by applied their proposed enhance GA. The enhanced GA is proposed by introducing a number of problem specific operators, including gene swap, gene cross swap, gene copy, gene inverse as well as gene min-max. Furthermore, Lee et al. (Lee et al., 1995) applied a modified simple GA (SGA) integrated with the successive linear programming technique to solve the ORPD and investment planning problem. Moreover, in the study of (Iba, 1994), the first application of GA for ORPD problem is presented by Iba in the year 1994. The proposed method break down the system into a number of sub-system and adopts interbreeding between the subsystems to produce new solutions.

2.4.1.4 Self-Adaptive Real Coded Genetic Algorithm (SARGA)

Apart from GA, self-adaptive real coded genetic algorithm (SARGA) is another approach that has been proposed for the solution to ORPD problem (Subbaraj & Rajnarayanan, 2009). As mentioned before, GA is a stochastic search optimization method. It is inspired by Darwinian principles of natural evolution. Undeniably, floating point expressions always superior than binary expressions due to their precision, consistency and rapid convergence for real valued engineering optimization problems (Michalewicz, 1996). Thence, in this study (Subbaraj & Rajnarayanan, 2009), Subbaraj & Rajnarayanan applied SARGA to ORPD problem by concerning discrete, continuous as well as binary variables. The self-adaptive characteristic is presented by applying the simulated binary crossover (SBX) operator which this phenomenon makes GA flexible and closer to natural evolution.

Based on the simulation results obtained in (Subbaraj & Rajnarayanan, 2009), SARGA proved its fast convergence and ability to converge to near global optimal solutions as compared to EP. Additionally, it also obtains a lesser loss reduction by less number of population size than PSO, MAPSO, GS, DE and hybrid EP. SARGA is further advantaged by its robustness. It is able to converge to near global optimum with higher probability percentage.

2.4.2 Physic-Based

Another category of techniques is those inspired by natural physical phenomena. Physic-based algorithms included gravitational search algorithm (GSA), charged system search (CSS), central force optimization (CFO), kinetic gas molecules optimization (KGMO), chemical reaction optimization (CRO), artificial chemical reaction optimization algorithm (ACROA) as well as black hole (BH) algorithm. Additionally, there are also other population-based techniques inspired by distinct sources which are under the category of physic-based algorithms: harmony search algorithm (HSA), improved HSA (IHSA), flower pollination algorithm (FPA), seeker optimization algorithm (SOA) and start of mater search (SMS) (Mirjalili, 2015a). In this section, HSA, IHSA, GSA and opposition-based GSA (OGSA) will be introduced and discussed as solutions for solving ORPD problem.

2.4.2.1 Harmony Search Algorithm (HSA) & Improved HSA (IHSA)

Harmony search algorithm (HSA) is a stochastic optimization algorithm which is developed based on the improvisation process of music players (Geem, Kim, & Loganathan, 2001). HSA has been successfully applied for solving a variety of optimization problems due to its basic advantages: The simplicity of its mathematical operations, its randomness in selecting the control variables and it does not require derivative operations as the whole searching process is a random process. Moreover, in this algorithm, all of the current vector solutions are considered in producing the elements of the new vector solution (Khazali & Kalantar, 2011).

Since 2011, Khazali and Kalantar introduced HSA in solving ORPD problems for determination of the global and near global minimum solution. According to the simulation results, it indicates that HSA is advantaged over SGA and PSO in decreasing transmission loss and voltage deviation as well as increasing voltage stability margin (Khazali & Kalantar, 2011). In year 2008, improved HSA (IHSA) or known as hybrid Taguchi HSA is developed and introduced to solve various engineering optimization problems (Yildiz, 2008) including OPF problem. IHSA adopts the Taguchi method to decrease the intervals of design criteria to obtain better initialization. It is found that both HSA and IHSA can obtain optimal results in a short computational time. Additionally, they can acquire near global minimum solutions by successfully preventing local minimum stagnation (Sinsuphan et al., 2013).

2.4.2.2 Gravitational Search Algorithm (GSA) & Opposition-based GSA (OGSA)

In year 2009, Rashedi et al. first introduced gravitational search algorithm (GSA) (Rashedi, Nezamabadi-Pour, & Saryazdi, 2009). It is developed based on the Newton's law of gravitation and the metaphor of gravitation interaction between masses. Prior to the year 2012, GSA has been proven its efficiency in overcoming OPF problems with different objective functions. GSA was successfully performed in obtaining the optimal results and the comparison between other methods again validates its superiority (Serhat Duman et al., 2012). The powerful characteristic of GSA is its gravitational constant. This characteristic helps to adjust the accuracy of the search, so it is to accelerate the optimization process (Ceylan, Ozdemir, & Dag, 2010; Rashedi et al., 2009; Rashedi, Nezamabadi-Pour, & Saryazdi, 2010). Moreover, GSA works efficiently like other techniques with memory, although it is memory-less (Rashedi et al., 2009, 2010).

In (Lee et al., 1995), Lee et al. applied a hybrid method between GSA and successive linear programming to overcome reactive power operation issue. Lately, in order to further improve the optimization performance of basic GSA in terms of its robustness and searching ability, opposition-based learning is implemented in the opposition-based gravitational search algorithm (OGSA) for generation jumping and population initialization. In this study (Shaw et al., 2014), OGSA is utilized for solving the ORPD problem. From the simulation results obtained, it is proven that the superiority and robustness of OGSA in solving ORPD problems as well as its capability in obtaining optimal results of the control variables of the tested bus system. As a conclusion, this study indicates that the proposed OGSA is able to obtain superior results as compared with the results of basic GSA formerly proclaimed in the recent up-to-date literature.

2.4.3 Swarm Intelligence (SI)

The third class of meta-heuristic methods is swarm intelligence (SI) algorithm. Some of the recently developed SI algorithms included artificial bee colony (ABC) (Karaboga & Basturk, 2007), cuckoo search (CS) (Yang & Deb, 2009), cuckoo optimization algorithm (COA) (Rajabioun, 2011), bat algorithm (BA) (Yang, 2010a), firefly algorithm (FA) (Yang, 2010b), fruit fly optimization algorithm (FFOA) (Pan, 2012), dolphin echolocation (DE) (Kaveh & Farhoudi, 2013), honey bee mating optimization (HBMO) and grey wolf optimizer (GWO) (Mirjalili, Mirjalili, & Lewis, 2014). Moreover, particle swarm optimization (PSO) and its variants also been categorized under the class of SI algorithm. In this section, SI algorithms including ABC, HBMO, PSO and variants of PSO will be further discussed and presented as solutions for solving ORPD problem.

2.4.3.1 Artificial Bee Colony (ABC)

Artificial bee colony (ABC) is a swarm-based search algorithm which was developed by Karaboga in the year 2005 (Karaboga, 2005). It is developed based on the motivation of the foraging nature of honeybee swarm. In ABC, the honeybee swarm is classed into mainly two categories which are worker bees and non-workers bees. Non-workers bees are further divided into explorer bees and onlooker bees. During the ABC process, the onlooker bees will be taken over by worker bees if the generated onlooker bees discover a source having better fitness value; else the base value is increased by 1. However, this source will be deserted if the base value outpaces its max boundary. The bee of the source is then regenerated with the explorer bee. Finally, the fittest bee will be represented in the coming iteration. The optimization process will be repeated until it is terminated by the stopping criterion (Ayan & Kılıç, 2012).

In year 2010, Ozturk et al. encountered ORPD problems by firstly applying ABC algorithm (Ozturk, Cobanli, Erdogmus, & Tosun, 2010). The simulation works are tested on IEEE 10-bus system. Also, other researchers, Ayan and Kılıç implemented ABC algorithm in solving ORPD problem. They tested its efficiency on IEEE 30-bus system and IEEE 118-bus system (Ayan & Kılıç, 2012). According to the simulation results, the superiority of ABC algorithm on solving large-scale ORPD problem is guaranteed. In that study, it can be concluded that the ABC algorithm is able to converge to global optimum better than SARGA, PSO and CLPSO.

2.4.3.2 Honey Bee Mating Optimization (HBMO) & Chaotic Parallel Vector Evaluated Interactive HBMO (CPVEIHBMO)

Honey bee mating optimization (HBMO) is another population-based meta-heuristic method that has been proposed by Haddad et al. in the year 2006 (Haddad, Afshar, & Mariño, 2006). It has been utilized in solving ORPD problem. However, HBMO suffers from local optima entrapment when facing problems with heavier constraints and multiple local optima. Additionally, it is extremely relies on the adjustment of its parameters. All these problems consequently cause HBMO to be suffering from premature convergence. So, to retrieve the drawbacks of HBMO in order to solve ORPD problems, the standard HBMO is improved and modified (A. Ghasemi et al., 2014).

In (A. Ghasemi et al., 2014), an improved HBMO technique, namely chaotic parallel vector evaluated interactive honey bee mating optimization (CPVEIHBMO) is introduced to overcome the multi objective ORPD problem. It is tested on IEEE 30-bus system, IEEE 57-bus system and IEEE 118-bus system. From the simulation results, it is proven that CPVEIHBMO able to obtain the optimal results of the control variables while satisfying the physical constraints. In the study, it is found that CPVEIHBMO provides more rapid and robust convergence than HBMO, HSA, PSO, GA, GSA and BBO. From the simulation outcomes, the capability to avoid local optima entrapment, the computational speed and its convergence precision are remarkably improved. Therefore, it is concluded that by combining chaotic local search with IHBMO, CPVEIHBMO is able to avoid the solutions being trapped in local optima due to the characteristics of chaos.

2.4.3.3 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is a swarm-based stochastic search algorithm which inspired by the natural behaviour of bird flocks or fish schooling (Eberhart & Kennedy, 1995). In PSO, a swarm of particles is retained where each particle in the swarm serves as a candidate solution. Each particle in PSO is affected by the fittest position of its neighbouring particles and its own fittest position in the search area. The two basic operations of PSO are velocity update and position update.

Undeniably, PSO is a popular optimization technique that has been applied by a number of researchers in solving ORPD problems due to its simplicity in implementation and concept, its robustness computation and efficiency in convergence. In (Zhao et al., 2005), a multi-agent based PSO algorithm has been proposed by Zhao et al. as a solution for ORPD problems. Furthermore, PSO was adopted by Yoshida et al. for reactive power and voltage control concerning voltage security assessment (Yoshida, Kawata, Fukuyama, Takayama, & Nakanishi, 2000). Also in (W. Zhang & Liu, 2008), Zhang and Liu applied a fuzzy adaptive PSO (FAPSO) for reactive power as well as voltage control.

Although PSO has gained attention from these researchers due to its searching capability as well as its novelty. However, PSO is easily being suffered from local optima stagnation when handling optimization problems with multiple local optima. Additionally, proper parameters setting will greatly influence the searching performance of PSO (Shaw et al., 2014; J. Zhang & Sanderson, 2009). Hence, local optima entrapment and premature convergence of PSO often occurr in numerous engineering optimization problems (Lampinen & Zelinka, 2000; Shaw et al., 2014).

2.4.3.4 Comprehensive Learning PSO (CLPSO)

Comprehensive learning PSO (CLPSO) is a variant algorithm of PSO where a learning strategy is employed in CLPSO. This characteristic allows it to encounter the premature convergence of standard PSO. CLPSO has been applied by Mahadevan and Kannan in solving ORPD problems (Mahadevan & Kannan, 2010). In the study, the algorithm was evaluated with three objectives: minimization of real power loss, enhancement of voltage profile and improvement of voltage stability on IEEE 30-bus system and IEEE 118-bus system. According from the results of the research, it is proven that CLPSO algorithm can overcome the drawback of standard PSO besides finding better solutions for ORPD problem.

2.4.3.5 Multiagent-based PSO (MAPSO)

Multiagent-based PSO (MAPSO) is a stochastic method that integrated multiagent system and PSO algorithm. MAPSO was proposed and developed by Zhao et al. for solving ORPD problems (Zhao et al., 2005). In MAPSO, all of the particles constructed in a lattice-like environment where each particle represents a candidate solution to the optimization problem. These particles compete and cooperate with their neighbouring particles in order to obtain the optimal solution. Thence, MAPSO is able to find better quality of solutions reliably with rapid convergence by their evolution mechanism and agent-agent interactions property. In this paper, MAPSO is applied in solving ORPD problem and evaluated on IEEE 30-bus system and a practical IEEE 118-bus system. From the results, it is concluded that MAPSO is able to obtain global optimum and it converged to better solutions faster than EP, PSO, SGA and adaptive GA (AGA).

2.5 Hybrid Methods

In (Huang & Huang, 2012), the authors proposed a hybrid technique by combining differential evolution and ant system (DE-AS) (Dorigo, Maniezzo, & Colorni, 1996) to solve ORPD problem. This algorithm provides higher percentage of getting global results than other reported methods based on its global search capability and fast convergence characteristic. The proposed approach is evaluated on IEEE 30-bus system and it achieved higher quality solutions than EP, PSO and DE by giving lower power transmission losses in short convergence speed.

In the research of (M. Ghasemi et al., 2014), a powerful algorithm, namely hybrid modified imperialist competitive algorithm and invasive weed optimization (MICA-IWO) is developed and introduced for overcoming ORPD problem. This algorithm has been tested on IEEE 30-bus system, IEEE 57-bus system and IEEE 118-bus system. The proposed algorithm is verified to be able to balance the global search capability and computational speed efficiently thru the simulation results. Moreover, it has also been proven to be better than imperialist competitive algorithm (ICA) and invasive weed optimization (IWO).

In the year 2006, Yan et al. proposed a hybrid method by combining genetic algorithm and interior point method (GA-IPM) as a solution to ORPD problems (Yan, Liu, Chung, & Wong, 2006). In this research, the ORPD problem is firstly being solved by utilizing IPM. Then, the second part comprises of two sub problems which are discrete and continuous optimization of ORPD problem. IMP is applied to solve the continuous nature while GA is employed to solve the discrete nature. Additionally, the efficiency of GA-IPM is further improved by the dynamic adjustment strategy of this technique.

2.6 Summary

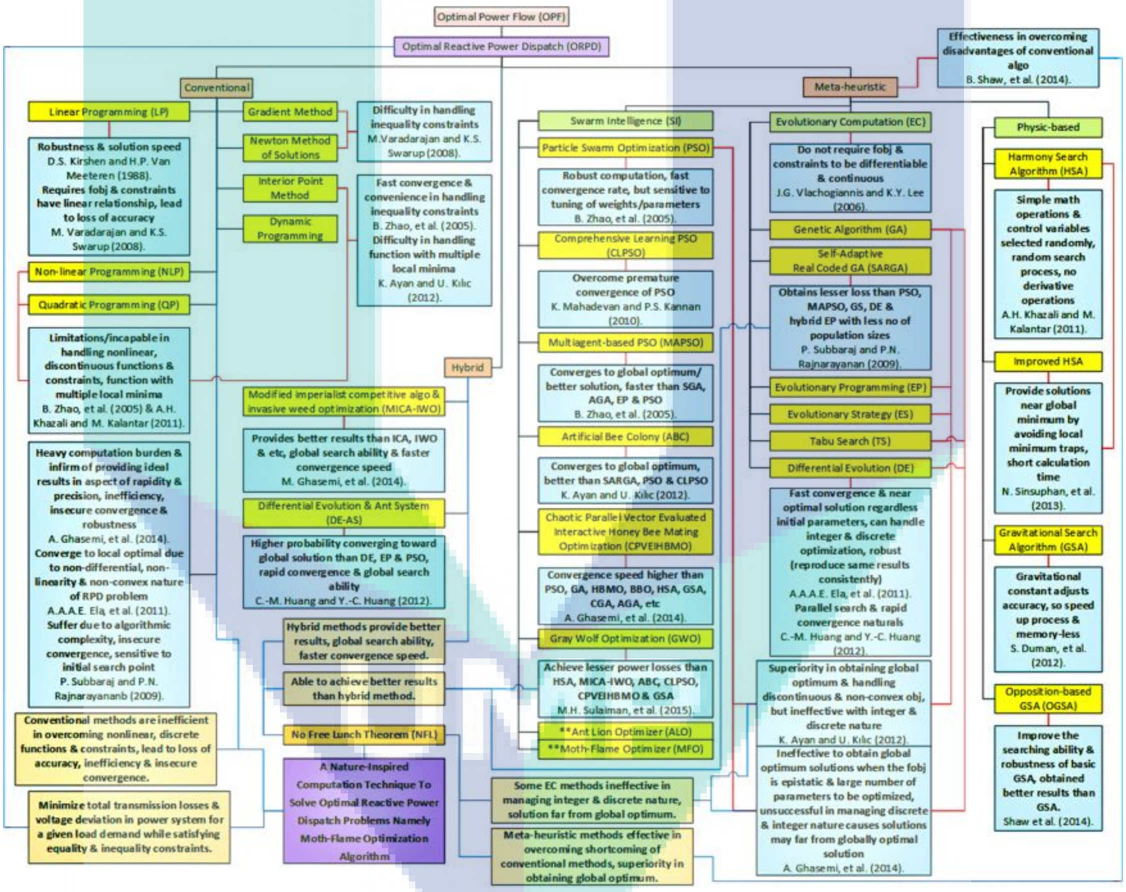

Figure 2.1 Literature map

From all the reviewed methods, many optimization methods including conventional, meta-heuristic and hybrid methods have been applied to solve the problem in ORPD. However, conventional methods failed to solve ORPD due to the nonlinearity, non-convex and non-differential nature of ORPD. They often converge to local optimal solutions which is far from the global optimum. This consequently causes loss in accuracy and insecure in convergence. Additionally, they need heavy computation burden due to their insufficient in searching ability and infirm of getting global best solutions preciously in the aspect of rapidity (El Ela et al., 2011; A. Ghasemi et al., 2014; Subbaraj & Rajnarayanan, 2009). All these drawbacks lead to the introduction and development of meta-heuristic optimization methods which ever to overcome the nonlinear characteristics of ORPD.

The meta-heuristic methods are divided mainly into three big groups based on their development inspiration: swarm intelligence (SI), evolutionary computation (EC) and physic-based. Physic-based stochastic search algorithms including HSA, IHSA, GSA and OGSA have been discussed in this project. In (Khazali & Kalantar, 2011), HSA is proven to be able to obtain better solutions than PSO and SGA in decreasing power loss, voltage deviation and improving voltage stability. Nevertheless, its results cannot beat CLPSO, SARGA, DE and SOA. Following it, the basic HSA is modified where improved HSA (IHSA) is introduced to solve OPF-based problem (Sinsuphan et al., 2013). Additionally, GSA physic-based algorithm which proposed by (S Duman, Sonmez, Guvenc, & Yorukeren, 2012) is also involved in solving OPF-based problem. It is able to improve the voltage profile, voltage stability and minimizing the cost function successfully with considering valve-point loading effect. It is better than PSO, DE, biogeography-based optimization (BBO) and etc. as reported in the study. Furthermore, in (Shaw et al., 2014), opposition-based GSA (OGSA) is proposed for the solution to ORPD in terms of minimizing power loss, voltage deviation and improving voltage stability index. It is developed to further strengthen the original GSA in terms of searching ability. The simulation results proved that OGSA yields better solutions than GSA, PSO, CLPSO, DE, SARGA, BBO, SOA and etc. with short simulation time.

Apart from that, EC is another category of heuristic methods that developed based upon the natural evolutionary process. In this project, the EC algorithms including EP, DE, GA and SARGA have been presented as a solution to ORPD problems. Years back to 1994 and 2010, GA has been applied for reactive power allocation planning (Iba, 1994) and voltage stability enhancement (Devaraj & Roselyn, 2010) in power system. However, Iba claimed that GA is weaker than conventional methods for large system in the aspect of computational speed. In (Subbaraj & Rajnarayanan, 2009), the researchers proved that SARGA, a variant technique of GA is able to obtain minimum power loss than PSO, MAPSO, DE, GS and hybrid EP.

Another popular EC-based algorithm, namely DE possess advantages in resolving ORPD via its parallel search and fast convergence characteristics. DE is validated to be superior than SQP, IPM, PSO, GA and etc. in terms of producing the lowest power loss (El Ela et al., 2011; Varadarajan & Swarup, 2008a). Additionally, it can obtain a lower voltage deviation than PSO (El Ela et al., 2011). Nevertheless, not all EC algorithm able to provide global optimum. Many EC methods such as GA, DE, EP, ES, TS and even PSO loss efficiency in handling problems with integers and discrete nature. In practice, this may consequently cause the solutions converge far from the global optimum (Ayan & Kılıç, 2012; A. Ghasemi et al., 2014). Furthermore, these methods demoted efficiency in obtaining global optimum when the parameters to be optimized is relatively large and highly correlated (A. Ghasemi et al., 2014).

Recently, many SI-based algorithms have been proposed to be implemented in the problems in ORPD. In this report, PSO, CLPSO, MAPSO, ABC, HBMO and CPVEIHBMO have been discussed. Both CLPSO and MAPSO are the variants of PSO, which developed to improve and overcome the premature convergence of original PSO (Mahadevan & Kannan, 2010; Zhao et al., 2005). Besides, MAPSO is found to be outperformed by its superiority in converging to global optimum in a fast computation rate than PSO, EP, SGA and AGA (Zhao et al., 2005). In (Ayan & Kılıç, 2012), ABC has proved to be better than PSO, CLPSO, SARGA, Enhanced GA, GS, DE, IPM and quantum-inspired EA (QEA) in minimizing power loss. Later, (A. Ghasemi et al., 2014) implemented their proposed CPVEIHBMO, a variant algorithm of HBMO in searching the optimal solution of a multi-objective ORPD problem with considering the physical constraints. The solutions obtained by CPVEIHBMO are outperforms than the solutions of HBMO, PSO, CLPSO, DE, GA, GSA, HSA and etc. The results obtained by employing ABC and CPVEIHBMO are obviously outperform than other methods. However, their results still can be beaten by recently developed algorithm, namely grey wolf optimizer (GWO) (Mohd Herwan Sulaiman et al., 2015). GWO is applied to solve ORPD by minimizing power loss and voltage deviation. Its results are better than HSA, MICA-IWO, ABC, CLPSO, CPVEIHBMO, GSA and etc.

On the other hand, hybrid methods also play significant role in successfully and efficiently in solving ORPD problem. In this report, three examples of hybrid methods have been presented. For instance, (Yan et al., 2006) applied their proposed hybrid

GA-IPM, which integrates the advantages of both IPM and GA. Its simulation outcomes illustrated that the proposed GA-IPM is better than the original GA and IPM in solving ORPD. In (M. Ghasemi et al., 2014), the researchers proposed MICA-IWO to solve ORPD and its simulation results are superior than ICA, IWO, PSO, CLPSO, MAPSO, SGA, DE, HSA and etc. Hereafter, (Huang & Huang, 2012) introduced hybrid DE-AS to minimize the power transmission loss and produce the lowest loss among PSO, EP and DE. Recent years, many hybrid methods have been proposed and applied to ORPD problem. For examples, these included hybrid PSO-GSA (Radosavljević, Jevtić, & Milovanović, n.d.), hybrid PSO-MVO (P. Jangir, Parmar, Trivedi, & Bhesdadiya, 2016), hybrid PSO-ICA (Mehdinejad, Mohammadi-Ivatloo, Dadashzadeh-Bonab, & Zare, 2016) and etc. Although hybrid methods provide better results that converge toward global solutions via their global search ability and rapid convergence speed, their results still can be taken over by newly developed meta-heuristic algorithms. Moreover, hybrid methods require more research studies upon the optimization algorithms and their application is much more complex than single heuristic methods.

Due to the increasing complexity of modern power system network in terms of functions and structures, this contributes to the demand for developing new meta-heuristic algorithms also consequently arise to improve ORPD problem in power system. The examples of these recently proposed heuristic methods involving Antarctic krill herd algorithm (AKHA) (Lenin, n.d.), improved baboon algorithm (IBA) (Lenin, Reddy, & Kalavathi, n.d.), fuzzy adaptive heterogeneous CLPSO (FAHCLPSO) (Naderi, Narimani, Fathi, & Narimani, 2017), modified DE algorithm (MDEA) (Sakr, EL-Sehiemy, & Azmy, 2017), quasi-oppositional CRO (QOCRO) (Dutta, Paul, & Roy, 2016), improved GSA-based conditional selection strategies (IGSA-CSS) (Chen, Liu, Zhang, & Huang, 2017), exchange market algorithm (EMA) (Rajan & Malakar, 2016), multi-verse optimizer (MVO) (Mohd Herwan Sulaiman, Mohamed, Mustaffa, & Aliman, 2016) and etc. Although the optimization algorithms aforementioned are able to overcome the problem of ORPD successfully and efficiently, the so-called no-free-lunch (NFL) theorem (Wolpert & Macready, 1997) allows more researchers to develop and propose new optimization algorithms which are more powerful in future time. According to NFL, the authors stated that there is no a single optimization technique that can solve all of the engineering optimization problems. Thence, NFL encourages us make a hypothesis that ORPD problem can yet to be solved by newly developed algorithms better than the present optimization techniques.

Undeniably, there are still some meta-heuristic algorithms which have not been implemented in ORPD to evaluate its performance efficacy in solving nonlinearities problem which involving discrete and continuous nature. In order to improve ORPD problem through minimization of transmission loss and voltage deviation, a new nature-inspired meta-heuristic algorithm, namely moth-flame optimization (MFO) algorithm is proposed. MFO not yet applied in ORPD and it is chosen because it is outperform other optimization techniques upon the majority of the case studies in (Mirjalili, 2015a) and real challenging constrained engineering design problems as reported in (N. Jangir et al., 2016; Trivedi, Bhesdadiya, et al., n.d.). In (Mirjalili, 2015a), MFO demonstrates rival exploration and exploitation in solving constrained optimization problems with uncertain search area. Furthermore, it can prevent local optima entrapment effectively due to its population-based optimization characteristic. In MFO process, the moths update their locations constantly according to the fittest flames obtained so far over the course of iteration, thus, this ensured the spiral convergence of MFO. Additionally, the best promising solutions of MFO are saved for each iteration that assists it will never get loss of its optimal solutions and the solutions will not be far from the global optimum (Allam, Yousri, & Eteiba, 2016; Mirjalili, 2015a). The test cases also validated MFO able to balance and control exploration and exploitation processes properly and effectively. MFO is simple and do not require much control parameters other than maximum iteration and size of population. As in (Bentouati, Chaib, & Chettih, 2016; N. Jangir, Pandya, Bhesdadiya, & Jangir, n.d.; Parmar et al., 2016; Trivedi, Parmar, et al., n.d.; Trivedi, Jangir, Parmar, & Jangir, n.d.), MFO has been applied for OPF and combined economic emission dispatch (CEED) problems while it has not applied yet to the problem of ORPD. Hence, MFO can be chosen as an alternative optimization technique for solving ORPD problems among the other present popular techniques.

CHAPTER 3

METHODOLOGY

3.1 Introduction

In this chapter, the proposed methodology applied in this research to achieve the targeted objectives within the scopes of study is presented. Firstly, the mathematical formulation of optimal reactive power dispatch (ORPD), moth-flame optimizer (MFO) and ant lion optimizer (ALO) will be deeply presented. Followed by the application of MFO and ALO in solving ORPD problem.

3.2 Mathematical Formulation of Optimal Reactive Power Dispatch (ORPD)

3.2.1 Objective Functions

In this project, the goals (objective functions) of ORPD problem are to minimize power transmission losses and voltage deviation of the power system while satisfying the physical and operating constraints. The problem of ORPD can be expressed as the minimization of function $f(x,u)$ as follows:

Minimize

$$f(x,u) \tag{3.1}$$

while subjected to

$$\begin{aligned} g(x,u) &= 0 \\ h(x,u) &\leq 0 \end{aligned} \tag{3.2}$$

where $f(x,u)$ is the objective function. In addition, $g(x,u)=0$ represents the equality constraints which are the power flow equalities whereas $h(x,u)\leq 0$ indicates the inequality constraints. The inequality constraints included generator bus voltage, transformer tap ratio and amount of reactive compensator. Also, x and u are the vector of dependent variables and control variables, respectively. As aforementioned, the first objective function of ORPD is the minimization of total system transmission loss, F_1 which it is an economic loss that do not give any profit or advantage to the society. The second objective function, minimization of voltage deviation, F_2 where this helps to improve the stability and security of the power system. F_1 and F_2 can be expressed as follows (Khazali & Kalantar, 2011; Mohd Herwan Sulaiman et al., 2015):

$$F_1 = P_{Loss}(x,u) = \sum_{k=1}^{Nl} g_k \left(V_i^2 + V_j^2 - 2V_iV_j \cos(\theta_i - \theta_j)\right) \qquad 3.3$$

where Nl indicates the transmission lines' number, and g_k represents the k th line conductance. V_i and V_j are the voltages at the end of bus-i and bus-j of the k th line, respectively. Besides, θ_i and θ_j are the line angles at the line i and j ends, respectively.

$$F_2 = VD(x,u) = \sum_{i=1}^{Nd} |V_i - V_i^{sp}| \qquad 3.4$$

where V_i implies the voltage at load bus-i, V_i^{sp} represents the specified value (usually equal to 1.0 p.u), and Nd denotes the load buses' number.

3.2.2 Equality Constraints

The equality constraints are the power balanced equations. The power equality of load flow declared that the total power losses equal to the total power generation minus the total load demands (Khazali & Kalantar, 2011). The equality constraints can be described by the equations as below:

$$P_{Gi} - P_{Di} = V_i \sum_{j \in N_i} V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}\right) \qquad 3.5$$

$$Q_{Gi} - Q_{Di} = V_i \sum_{j \in N_i} V_j \left(B_{ij} \cos\theta_{ij} - G_{ij} \sin\theta_{ij} \right) \quad 3.6$$

where V_i and V_j denote the voltages at load bus-*i* and bus-*j*, respectively. B_{ij} is the susceptance between bus-*i* and bus-*j* whereas G_{ij} is the conductance between bus-*i* and bus-*j*. P_{Gi} and Q_{Gi} are the real power generation and reactive power generation, respectively. On the contrary, P_{Di} and Q_{Di} are the real load demand and reactive load demand, respectively.

3.2.3 Inequality Constraints

In ORPD, the inequality constraints involving the constraints of generators, transformers and reactive devices. As aforementioned, ORPD is a problem that comprises of continuous and discrete parameters. In order to combat the discrete variables, they are considered as continuous variables at the beginning of the optimization. At the end of the optimization, the continuous values are mapped back to the discrete values. In this project, four decimal places are kept at the end of the optimization. Furthermore, all the inequality constraints must be limited by their lower and upper boundaries so that stable operation can be attained. Firstly, the generators constraints included real power generation, reactive power generation and bus voltage generation must be restricted by their lower and upper bounds as below:

$$P_{Gi}^{\min} \leq P_{Gi} \leq P_{Gi}^{\max} \quad 3.7$$

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max} \quad 3.8$$

$$V_{Gi}^{\min} \leq V_{Gi} \leq V_{Gi}^{\max} \quad 3.9$$

where $i = 1, \ldots, N_G$ and N_G is the generators' number.

Secondly, the setting of the transformers tap ratio must be restricted by their ranges as follows:

$$T_i^{\min} \le T_i \le T_i^{\max} \qquad 3.10$$

where i = 1, …, N_T and N_T is the transformers' number.

On the other hand, the reactive compensators sizing is limited within their minimum and maximum ranges as below:

$$Q_{Ci}^{\min} \le Q_{Ci} \le Q_{Ci}^{\max} \qquad 3.11$$

where i = 1, …, N_C and N_C is the reactive compensators' number.

3.2.4 Penalty Function Method

ORPD is defined as a constrained problem where the inequality constraints including voltage of generator buses, tap ratio of transformers and sizing of reactive compensators are control parameters which are self-constrained. On the other hand, the equality constraints (power balanced equations) in this research will be automatically satisfied by utilizing MATPOWER. Thence, the voltage magnitude of *PQ*-buses as well as the injected reactive power generation of *PV*-buses are constrained by taking them into account as penalty terms to the objective function (F_1 or F_2). The above issue can be formulated by the equation below (Li, Cao, Liu, Liu, & Jiang, 2009; Mallipeddi, Jeyadevi, Suganthan, & Baskar, 2012; Y. Zhang & Ren, 2004):

$$F = F_X + \sum_{i \in N_V^{\lim}} \alpha_{Vi} (V_i - V_i^{\lim})^2 + \sum_{i \in N_Q^{\lim}} \alpha_{G_i} (Q_{Gi} - Q_{Gi}^{\lim})^2 \qquad 3.12$$

F is the fitness function comprises of objective function and penalty function terms, where F_X is the objective function. N_V^{lim} represents the set of number of buses violating the voltage magnitude limits. On the contrary, N_Q^{lim} represents the set of number of buses violating the injected reactive power limits. In addition, α_{Vi} and α_{Gi} are the penalty factors. On the other hand, both V_i^{lim} and Q_{Gi}^{lim} are defined as:

$$V_i^{\lim} = \begin{cases} V_i^{\max}; V_i > V_i^{\max} \\ V_i^{\min}; V_i < V_i^{\min} \end{cases} \qquad 3.13$$

$$Q_{Gi}^{\lim} = \begin{cases} Q_{Gi}^{\max}; Q_{Gi} > Q_{Gi}^{\max} \\ Q_{Gi}^{\min}; Q_{Gi} < Q_{Gi}^{\min} \end{cases} \quad 3.14$$

If the control parameters exceed the range of the limits, the penalty function would then be taken into account to the objective function to combat the violation. On the other hand, the penalty function would equal to zero when all the control parameters are in the range of their boundaries. Furthermore, in MATLAB programming, the penalty factor has often being assigned a big value. In this research, the penalty factor is set to 100000. The constrained ORPD problem will then convert to unconstrained ORPD problem due to the big values of penalty factor which continue increasing until approximating infinity. Moreover, in ORPD, the control variables as aforementioned need to be analysed and considered carefully besides avoid them exceed the voltage limits. Otherwise, it will cause significant damage to the transmission systems.

3.3 MATPOWER

In this project, it is vital to highlight that MATPOWER 5.1 software toolbox (Zimmerman, Murillo, x, nchez, & Thomas, 2011) has been introduced and implemented in order to satisfy the equality constraints, obtain the targeted objective functions and calculate the power flow. This toolbox is applied in this research to ensure precise and accurate solutions can be achieved by executing the load flow program. MATPOWER comprises of a package of MATLAB M-files which developed to provide the best performance possible. The beauty of MATPOWER is its simplicity in coding which easy to be understood, customize and modify.

MATPOWER 5.1 consists of four different power flow solvers for solving a load flow analysis. These solvers included Newton-Raphson method, variations of fast-decoupled methods (XB and BX fast-decoupled) and Gauss Seidel. However, the default algorithm to execute the load flow problem is based upon Newton's method utilizing a full Jacobian updated at each iteration (Ray & Murillo-Sanchez, 2015). In (Akorede & Hizam, 2009), the authors claimed that the solutions of the power flow problem executed by MATPOWER are almost exactly the same as the one obtained by Saadat's method using MATLAB (Saadat, 1999). Thus, this fact ensures the precision of MATPOWER software package for solving load flow analysis.

This research implements MATPOWER 5.1 with MFO in solving ORPD in order to obtain the global optimum. In this project, the results of the compared algorithms also being mapped into the same MATPOWER load flow program in order to get a fair comparison with the proposed algorithm.

3.4 Moth-Flame Optimizer (MFO)

Moth-flame optimization (MFO) algorithm is a nature-inspired meta-heuristic optimization algorithm developed by Seyedali Mirjalili (Mirjalili, 2015a). It is being proven to be competitive with other renowned optimization algorithms as stated in Mirjalili's paper. Moths are insects which are closely related to the butterflies' family. During their lifetime, they generally undergo two main milestones: larvae and adult stages. The inspiration of MFO algorithm is the unique navigation technique of moths at night time. The moths utilized a mechanism known as transverse orientation when navigate at night depending on the moonlight. They fly by retaining their position at a fixed angle with respect to the moon as illustrated by Figure 3.1.

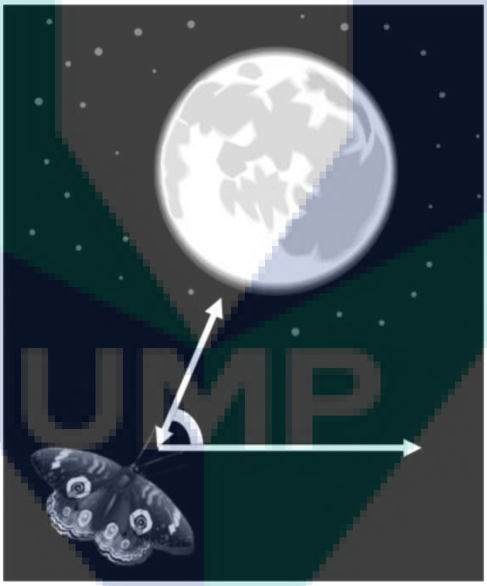

Figure 3.1 Transverse orientation of moth
Source: (Mirjalili, 2015a)

In nature, the moon is relatively far away from the moths and the moths are actually travelling in a straight line using transverse orientation. Therefore, this mechanism only helpful and useful for travelling in straight line when the source of light is extremely far. However, the moths in fact are mostly tricked by man-made light sources and fly spirally around the lights. In Addition, the moths also try to retain the similar angle with respect to the artificial light source. Nevertheless, this behaviour causes deadly spiral fly path for them as the light source is extremely close compared to the moon (Frank, Rich, & Longcore, 2006; Mirjalili, 2015a). The natural behaviour of the spiral flying path of moths is shown in Figure 3.2.

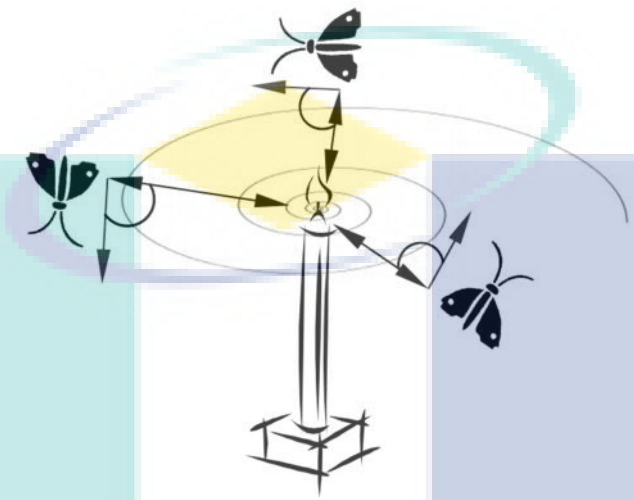

Figure 3.2 Moth's spiral flying path around close light source
Source: (Mirjalili, 2015a)

3.4.1 Mathematical Formulation of MFO

In MFO, moths are the candidate solutions and the position of moths in the search area is the problem's variables. MFO algorithm is a swarm-based algorithm. In order to model this algorithm, the first key component is the set of moths which can be expressed in a matrix as below:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & m_{n,d} \end{bmatrix} \qquad 3.15$$

where n is the number of moths, and d indicates the dimension (number of variables). The second key component of MFO is the set of flames which can be expressed in a matrix similar to the matrix of moth, M as below:

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & F_{n,d} \end{bmatrix} \qquad 3.16$$

where n and d are the number of moths and dimension, respectively. This matrix always stores n recent best results attained so far. During optimization, the moths are obliged to update their positions according to this matrix. Thus, the next position of a moth is defined based upon a flame in matrix F. Since the dimension of Eq. 3.15 and Eq. 3.16 are equal, it is assumed that there are arrays for storing the corresponding fitness values of the moths and the flames as below:

$$OM = \begin{bmatrix} OM_1 \\ \vdots \\ \vdots \\ OM_n \end{bmatrix} \qquad 3.17$$

$$OF = \begin{bmatrix} OF_1 \\ \vdots \\ \vdots \\ OF_n \end{bmatrix} \qquad 3.18$$

where n represents the number of moths. The fitness values are the return values or outputs of the objective function assigned for each moth and each flame. It is important to notice that moths and flames both are solutions. However, both of them are different in the aspect of the way to treat and update them. In MFO, flames are the best position of moths attains so far, whereas moths are the actual search agents that travel around the search area. Thus, flames are considered as flags that are dropped by the moths when searching the search area. Each moth searches around a flame and updates its position in order to find a better result. This mechanism helps the moth not to lose its best solution. The position of each moth updated according to a flame can be mathematically modelled by the equation below (Mirjalili, 2015a):

$$M_i = S\left(M_i, F_j\right) \qquad 3.19$$

where M_i and F_j represent the *i*-th moth and *j*-th flame, respectively. *S* indicates the spiral function. The following mathematical expression is the logarithmic spiral equation which it is the main update mechanism of moths (Mirjalili, 2015a):

$$S(M_i, F_j) = D_i \cdot e^{bt} \cdot \cos(2\text{®}t) + F_j \qquad 3.20$$

where *b* denotes a constant for defining the shape of the logarithmic spiral, and *t* denotes a random number within the interval of [-1,1]. Variables *t* indicates how close the next position of moth to the flame. It is worth to mention that each moth is obliged to update its location utilizing only one of the flames in this equation in order to avoid local optima entrapment. D_i denotes the distance of the *i*-th moth for the *j*-th flame where it can be calculated by the equation below:

$$D_i = | F_j - M_i | \qquad 3.21$$

where F_j represents the *j*-th flame whereas M_i represents the *i*-th moth. The main component of this algorithm is the spiral movement of the moths which decide the way they update their positions around the flames. Since Eq. 3.20 allows a moth to fly spirally around a flame and not essentially within area between them. Therefore, this equation ensures the exploration and exploitation processes can be achieved. Exploration happens when the next position of moth lies outside the area between the flame and moth whereas exploitation occurs when the next location of moth situates in the area between the flame and moth.

After updating the lists of flames, the flames are sorted and arranged according to their fitness values for each iteration. Then, the moths updated their locations based on their corresponding flames. Moreover, it is vital to note that in the initial stages of iterations, there are *N* number of flames. However, the number of flames will decrease gradually over the course of iterations. Hence, in the final stages of iterations, the moths update their locations only according to the fittest flame (the best flame). The decrement in number of flames helps to balance the exploitation and exploration of the search area (Mirjalili, 2015a). The following formula is expressed for the number of flames regarding this phenomenon:

$$flame\ no = round\,(N - l * \frac{N-1}{T}) \qquad 3.22$$

where N indicates the maximum number of flames. l indicates the current number of iterations whereas T is the maximum iterations.

3.4.2 Implementing MFO in ORPD Problem

The implementation of MFO algorithm in solving ORPD problem is via the finding of the optimal setting of control variables in order to minimize the objective function (total transmission loss, F_1 or voltage deviation, F_2) while satisfying both equality and inequality constraints. Firstly, the maximum number of iterations and the number of search agents are set. MFO algorithm is a population-based algorithm and moths are the candidate solutions for ORPD problem. The vector of the population is expressed in matrix as in Eq. 3.15. The row of the population matrix represents the search agents (moths) whereas the column represents the control variables (position of moths).

During the evaluation process, the position of each moth is mapped into the load flow data. Then, the load flow program is executed in order to obtain the transmission loss (Eq. 3.3). In each iteration, the moths update their positions with respect to their corresponding flames (utilizing Eqs. 3.19 - 3.21). After updating their positions, the transmission loss is obtained for the corresponding moth. The solutions are then sorted and arranged according to their fitness values and saved in matrix form. The best solution will be located in the upper part of the matrix while the worst solution will be located in the lower part of the matrix.

Furthermore, the updated control variables obtained so far will be checked if they are out of bounds from the constraints. If the control variables are out of bound, they will be tagged at the lower and upper boundaries. This is to ensure the obtained results are accurate and precise. The MFO algorithm will continue the process until it is terminated by the predefined stopping criterion (maximum iteration). Moreover, it is vital to mention that the voltage magnitude of each load bus must be in a specified range, for instance ±10 % is used in this research.

For second objective function (Eq. 3.4), the same procedures are utilized to evaluate the voltage deviation at load buses. The overall process of MFO in solving ORPD problems as explained above is simplified by Figure 3.3. The main simulation process of MFO algorithm in overcoming the problem of ORPD is depicted by the black dashed rectangular in Figure 3.3. Furthermore, the optimization process of MFO is explained by the pseudo code in Figure 3.4.

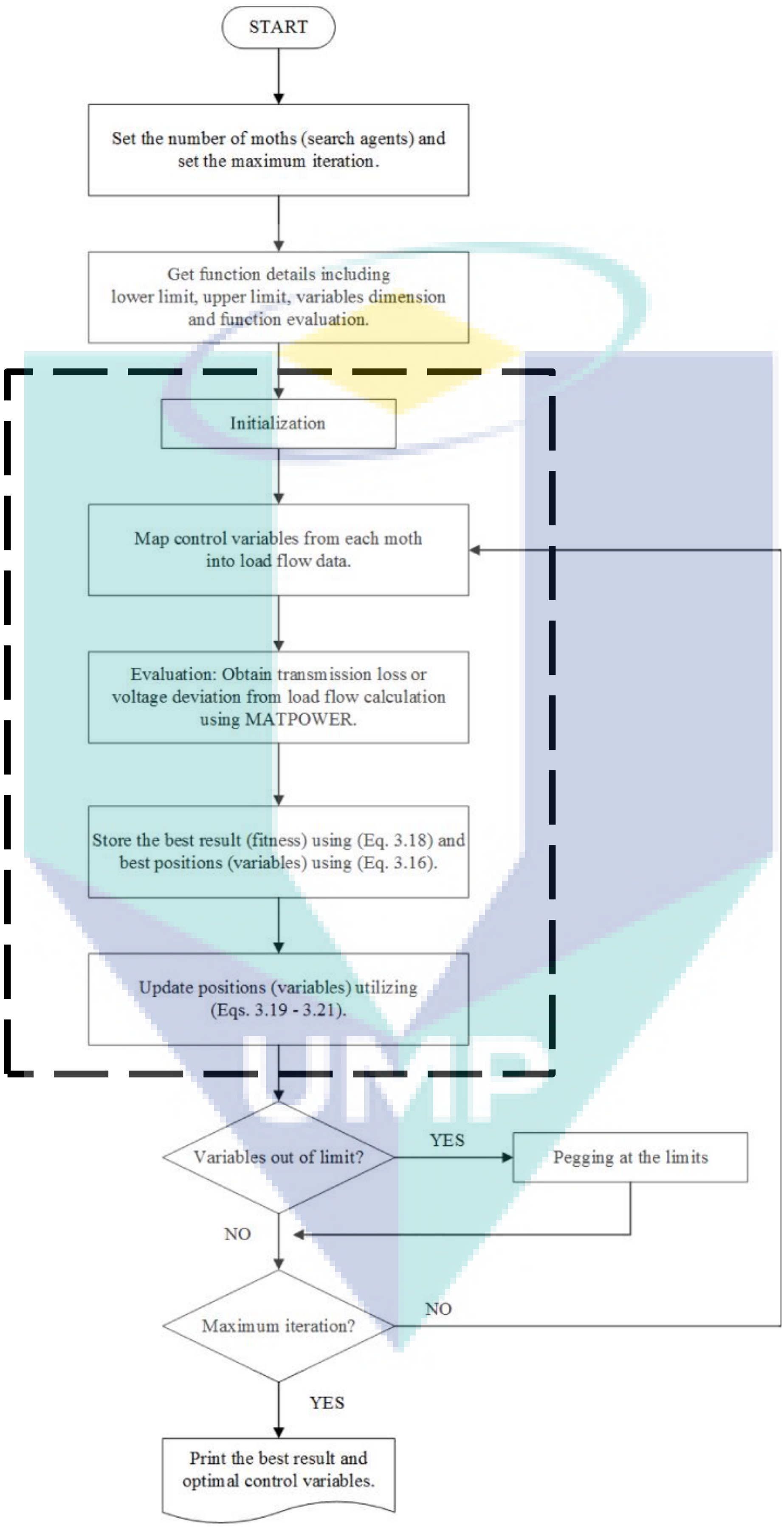

Figure 3.3 Flowchart of MFO algorithm for solving ORPD problem

```
begin
initialize the positions of moths
while ( iteration < max iteration )
update the number of flames according to Eq. 3.22
for ( i=1:size (moth_pos,1) )       //moth_pos: position of moth
check if moths go out of the search space
if the moth is out of the search space, bring it back
calculate the fitness of the moths
end
if ( iteration==1 )
sort the first population of the moths
update the flames
else
sort the moths
update the flames
end
update the position of the best flame obtained so far
for ( i=1:size (moth_pos,1) )
    for ( j=1:size (moth_pos,2) )
        if  ( i <= flame_no )       //flame_no: number of flame
        update the position of the moth with respect to its corresponding flame
        using Eqs. 3.19 – 3.21
        end
        if ( i > flame_no )
        update the position of the moth with respect to only one flame (fittest flame)
        end
    end
end
print the best optimum results obtained so far
end
```

Figure 3.4 Pseudocode of MFO algorithm

3.5 Ant Lion Optimizer (ALO)

Ant lion optimizer (ALO) is first developed by Seyedali Mirjalili in the year 2015 (Mirjalili, 2015b). It is inspired by the foraging mechanism of antlions in catching preys. ALO is developed based upon five main stages: random walk of ants, entrapment of ants, building traps, catching preys and rebuilding traps. Firstly, the inspiration of ALO will be discussed followed by the mathematical modelling of ALO.

The name of antlions is initiated by their hunting behaviour and their preferably prey (ant). Antlions are insects that belong to the family of Myrmeleontidae and they undergo two main lifecycles: larvae and adult. Their hunting period mostly occurs during larvae and adult stages with the purpose for reproduction. During hunting, an antlion digs a cone-shaped trap and hides underneath the bottom of the trap. They waiting for preys to be trapped in the trap as illustrated in Figure 3.5. Once the prey is in the trap, the antlion will throw sand outward the trap to slide the prey toward it. The prey is pulled under the sand and the antlion consumed it. Then, the antlion rebuild the trap and waiting for the next hunt.

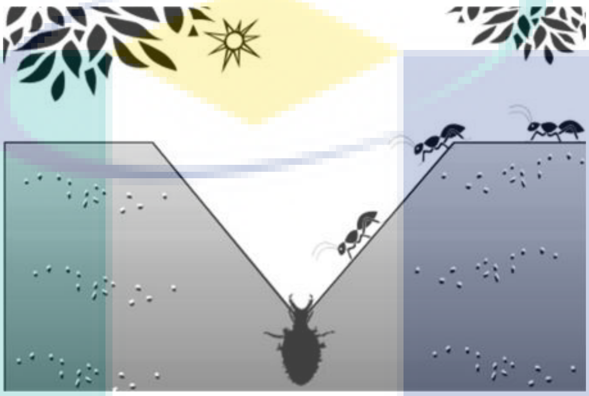

Figure 3.5 Cone-shaped pit and hunting mechanism of antlion
Source: (Mirjalili, 2015b)

3.5.1 Mathematical Formulation of ALO

a) Random walk of ants:

In ALO, ants are needed to move around the search area while antlions are allowed to catch them and become fitter using pits. ALO algorithm mimics the interaction between antlions and ants during hunting. In nature, ants move randomly over the search area when searching for food. This ants' stochastic movement can be mathematically being described as below (Mirjalili, 2015b):

$$X(t) = [0, cumsum(2r(t_1)-1), cumsum(2r(t_2)-1), \ldots, cumsum(2r(t_T)-1)] \quad 3.23$$

where t and T indicate the iteration (step of random walk) and maximum number of iterations, respectively. *cumsum* computes the cumulative sum, and $r(t)$ indicates the stochastic function expressed as below:

$$r(t) = \begin{cases} 1 & if \ rand > 0.5 \\ 0 & if \ rand \leq 0.5 \end{cases} \quad 3.24$$

where *rand* indicates a random number within the interval of [0,1]. During optimization, the ants' positions are used and saved in the form of a matrix as follows:

$$M_{Ant} = \begin{bmatrix} A_{1,1} & A_{1,2} & \cdots & A_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ A_{n,1} & A_{n,2} & \cdots & A_{n,d} \end{bmatrix} \quad 3.25$$

where M_{Ant} shows the matrix for saving the position of each ant. Parameters n and d are the number of ants and number of variables (dimension), respectively. $A_{i,j}$ denotes the value of j-th variable of i-th ant. During optimization, an objective function is used to evaluate the fitness of each ant and their fitness values are stored in the matrix as follows:

$$M_{OA} = \begin{bmatrix} f([A_{1,1}, & A_{1,2}, & \cdots, & A_{1,d}]) \\ & \vdots & & \\ & \vdots & & \\ f([A_{n,1}, & A_{n,2}, & \cdots, & A_{n,d}]) \end{bmatrix} \quad 3.26$$

where M_{OA} demonstrates the matrix for saving the fitness of each ant. n indicates the number of ants whereas d indicates the number of variables. $A_{i,j}$ indicates the value of j-th dimension of i-th ant and f indicates the objective function. In addition, the antlions are also hiding in traps somewhere in the search area which their positions and fitness values can be saved in the matrices as follows:

$$M_{Antlion} = \begin{bmatrix} AL_{1,1} & AL_{1,2} & \cdots & AL_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ AL_{n,1} & AL_{n,2} & \cdots & AL_{n,d} \end{bmatrix} \quad 3.27$$

where $M_{Antlion}$ shows the matrix for saving the position of each antlion. n denotes the number of antlions and d denotes the dimension. $A_{i,j}$ denotes the value of j-th variable of i-th antlion.

$$M_{OAL} = \begin{bmatrix} f([AL_{1,1}, & AL_{1,2}, & \cdots, & AL_{1,d}]) \\ & \vdots & & \\ & \vdots & & \\ f([AL_{n,1}, & AL_{n,2}, & \cdots, & AL_{n,d}]) \end{bmatrix} \quad 3.28$$

where M_{OAL} demonstrates the matrix for saving the fitness of each antlion. Variables n and d indicate the number of antlions and dimension, respectively. $A_{i,j}$ represents the j-th dimension's value of i-th antlion and f represents the objective function. Furthermore, ants will update their positions during each optimization. In order to keep the random walk of ants within the search bounds, Eq. 3.23 is normalized using the min-max normalization formula as follows:

$$X_i^t = \frac{\left(X_i^t - a_i\right) \times \left(d_i - c_i^t\right)}{\left(d_i^t - a_i\right)} + c_i \quad 3.29$$

where a_i represents the minimum of random walk of i-th variable. Parameters c_i and d_i are the minimum and maximum of all variables for i-th ant, respectively. On the other hand, variables c_i^t and d_i^t are the minimum and maximum of i-th variable at t-th iteration, respectively.

b) Trapping in antlions' traps:

The effect of antlions' traps on random walk of ants is mathematically formulated as follows (Mirjalili, 2015b):

$$c_i^t = Antlion_j^t + c^t \quad 3.30$$

$$d_i^t = Antlion_j^t + d^t \quad 3.31$$

where $Antlion_j^t$ represents the position of the selected j-th antlion at t-th iteration. Parameters c^t and d^t indicate the minimum and maximum of all variables at t-th iteration, respectively. Moreover, vectors c and d in Eq. 3.30 and Eq. 3.31 defined that ants move randomly in a hyper sphere around selected antlion as illustrated in Figure 3.6.

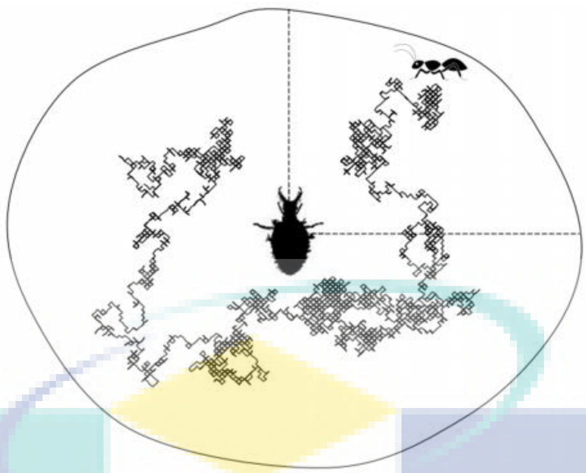

Figure 3.6 Random walk of ant in antlion's trap
Source: (Mirjalili, 2015a)

c) Building pits:

In ALO, ants are supposed to be captured by only one selected antlion. Thus, ALO algorithm implemented roulette wheel operator for choosing antlions based upon their fitness during optimization. This operator provides a higher opportunity to the fitter antlions for trapping ants.

d) Sliding ants against towards antlions:

Once the antlion realizes a prey is in the trap, it will try to catch the prey by shooting and throwing the sand outwards the centre of the trap. This behaviour slides down the trapped prey that is attempting to escape from the trap. This mechanism can be mathematically described by the equations below where the radius of hyper sphere is decreased adaptively (Mirjalili, 2015b):

$$c^t = \frac{c^t}{I} \qquad 3.32$$

$$d^t = \frac{d^t}{I} \qquad 3.33$$

where I is the ratio. $I = 10^w \frac{t}{T}$, t and T are the current iteration and maximum iterations, respectively. w is a constant that defined the accuracy level of exploitation.

e) Catching preys and rebuilding the traps:

Finally, the prey becomes fitter (goes deeply in the sand) than its corresponding predator when an ant is being caught by antlion in the trap. The antlion will consequently update its current position to the position of the hunted ant to improve its opportunity of catching new prey. The following formula described this regard (Mirjalili, 2015b):

$$Antlion_j^t = Ant_i^t \quad if \;\; f\left(Ant_i^t\right) > \left(Antlion_j^t\right) \tag{3.34}$$

where Ant_i^t and Ant_j^t represent the position of the selected *i*-th and *j*-th ant at *t*-th iteration, respectively. In addition, $Antlion_j^t$ represents the position of the selected *j*-th antlion at *t*-th iteration.

f) Elitism:

Elitism is a vital behaviour that helps ALO algorithm to maintain the best results attained for each optimization process. The fittest antlion obtained so far in each iteration is assumed as the elite. The elite is able to affect the movements of all the ants. Hence, all the ants randomly walk around a selected antlion by the roulette wheel and the elite simultaneously as below (Mirjalili, 2015b):

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{3.35}$$

where R_A^t and R_E^t are the random walk around the antlion selected by the roulette wheel and the elite at *t*-th iteration, respectively. Besides, Ant_i^t is the position of the *i*-th ant at *t*-th iteration.

3.5.2 Implementing ALO in ORPD Problem

The application of ALO algorithm in overcoming ORPD problem involves the searching of the optimal values of control variables to minimize the objective function (F_1 or F_2) while satisfying all the aforementioned constraints. Initially, both the maximum number of iteration and search agents' number (number of ants) are set. The ants are saved in a matrix such as in Eq. 3.25 where the row and column indicate the number of ants and number of variables, respectively.

During simulation process, each position of ants is mapped into the load flow data of MATPOWER using MATLAB. Then, the program of load flow is run to obtain the transmission loss. In each iteration, the obtained fittest antlion with minimum loss and its positions (variables) are stored. The fittest antlion is assumed as the elite. Then, the positions and fitness of antlions are updated based upon its corresponding ants. Followed by updating the position of the elite.

The evaluation process continues by evaluating ALO procedures according to Eqs. 3.23 – 3.35. The optimization is repeated until it is stopped by the predefined maximum iteration. On the other hand, the same steps of ALO algorithm are applied to evaluate the second objective function (minimization of voltage deviation). The implementation of ALO algorithm in overcoming ORPD problem is illustrated in the flowchart of Figure 3.7.

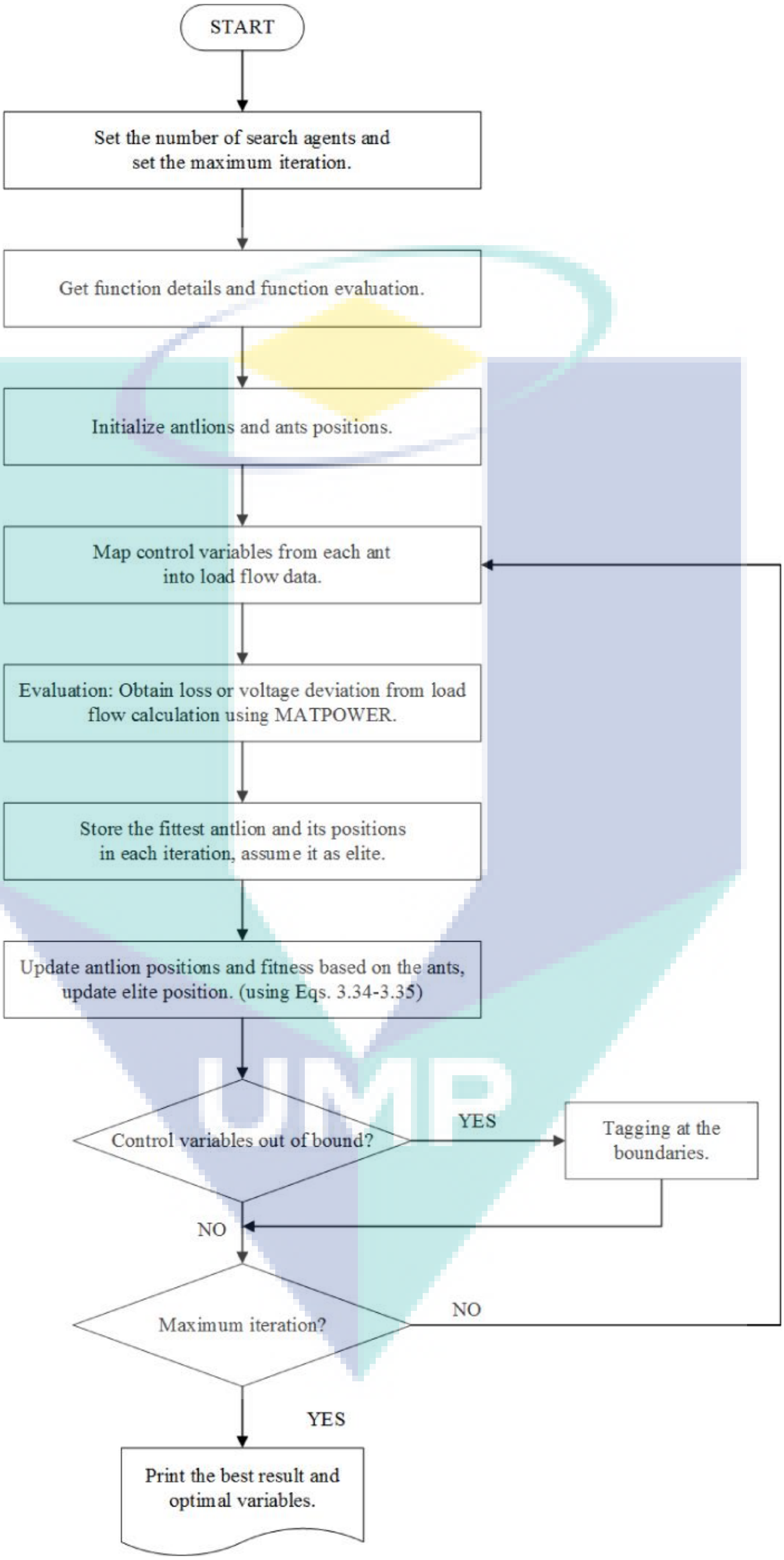

Figure 3.7 Flowchart of ALO algorithm for solving ORPD problem

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

The simulation results that obtained in this research is further discussed in this chapter in order to show the effectiveness of moth-flame optimization (MFO) algorithm on solving optimal reactive power dispatch (ORPD) problems. Both of the equality and inequality constraints are taken into account in order to make the ORPD problem more realistic. In this research, another algorithm namely ant lion optimizer (ALO) also has been applied as an alternative method in overcoming ORPD problem. Undeniably, many of the optimization algorithms are in fact designed from well-known and well-recognized meta-heuristic techniques such as differential evolution (DE), genetic algorithm (GA) and particle swarm optimization (PSO). Therefore, to assess the results of proposed MFO for ORPD problems, it is compared with those from the best known methods. In this research, DE and GA are chosen to be run by using MATLAB and MATPOWER to get the optimal results. On the other hand, the results of PSO are obtained from the selected paper and their results are compared with the proposed MFO.

MATLAB R2010a and MATPOWER 5.1 software package are utilized simultaneously in this research to execute the simulations. There are three test systems with five different case studies, including IEEE 30-bus system, IEEE 57-bus system and IEEE 118-bus system, are implemented in this study to prove and validate the efficiency of proposed MFO on solving ORPD problems. For each case study, different load demands (including active power and reactive power) have been set in order to observe the stability of the proposed algorithm. Furthermore, the statistical results obtained using MFO are compared against the results of ALO and the selected optimization algorithms as reported in the literatures. Table 4.1 summarizes the detail description of different case studies and Table 4.2 presents the overview of all the test system characteristics used in this research.

Table 4.1 Description of different case studies

Case Studies	Number of Control Variables	Load Demands	
		Active Power, P_{Load} (MW)	Reactive Power, Q_{Load} (MVar)
IEEE 30-Bus System	13	283.2	126.2
IEEE 30-Bus System	19	283.2	126.2
IEEE 30-Bus System	25	283.2	126.2
IEEE 57-Bus System	25	1250.8	336.4
IEEE 118-Bus System	77	4242.0	1438.0

Table 4.2 Overview of different test system characteristics

Descriptions	IEEE 30-Bus System	IEEE 57-Bus System	IEEE 118-Bus System
Test system type	small	medium	large
Number of busses	30	57	118
Number of branches	41	80	186
Number of generators	6	7	54
Number of transformers	4	15	9
Number of shunts	3 (test case 1) 9 (test case 2&3)	3	14
Number of equality constraints	60	114	236
Number of inequality constraints	125	245	572
Number of control variables	13 (test case 1) 19 (test case 2) 25 (test case 3)	25	77
Number of discrete variables	6	20	21

4.2 Test Case 1: IEEE 30-Bus System with 13 Control Variables

Initially, case study of IEEE 30-bus system consists of 13 control variables to be optimized will be tested. This test system is designed based upon (Khazali & Kalantar, 2011). It consists of 41 lines, six generators voltage and four tap changers where the transformers are located at lines 6-9, 6-10, 4-12 and 27-28. On the other hand, three reactive compensators are situated at buses 3, 10 and 24, respectively. The load demands for IEEE 30-bus system is set as

$$S = P + jQ = 2.832 + j1.262 p.u. \quad 4.1$$

where the system load is 283.2 MW and the initial loss is 5.663 MW. Additionally, the control variables are modelled as both continuous and discrete variables. The generator buses voltage, transformers tap setting and reactive compensators sizing are all constrained by their minimum and maximum limits as summarized in Table 4.3. The operating range of all tap changers is set in the interval of [0.95, 1.05] with a discrete step size of 0.01. Moreover, the boundary setting of all the control variables is set in a range of ±10 % in order to fulfil the operating constraints. For this case study, the number of function evaluations (NFE) to reach the best optimal results are 3000 and 4500 by using 20 search agents and 30 search agents, respectively. Table 4.4 presents the setting of the simulation parameters for DE, GA, ALO and MFO used in this research.

Table 4.3 Setting of variables limit for test case 1, IEEE 30-bus system

Control Variables	**Lower Bound**	**Upper Bound**
Generator Buses Voltage	0.90 p.u	1.10 p.u
Transformers Tap Setting	0.95 p.u	1.05 p.u
Reactive Compensators Sizing	-12 MVar	36 MVar

Table 4.4 Simulation parameters for DE, GA, ALO and MFO (test case 1)

Simulation Parameters	**DE** (Price, Storn, & Lampinen, 2006)	**GA** (Blasco, 1999)	**ALO**	**MFO**
Number of search agents/populations	30	30	30	20 (for MFO#20) 30 (for MFO#30)
Maximum number of iterations	150	150	150	150
Number of function evaluation, NFE	4500	4500	4500	3000 (for MFO#20) 4500 (for MFO#30)
Step size for DE, F	0.2	N/A	N/A	N/A
Crossover rate for DE, CR	0.6	N/A	N/A	N/A
Crossover probability for GA, Pc	N/A	0.9	N/A	N/A
Mutation probability for GA, Pm	N/A	0.01	N/A	N/A

For fair comparison between MFO against other chosen heuristic algorithms, the optimization results reported in (M. Ghasemi et al., 2014; Khazali & Kalantar, 2011; Mohd Herwan Sulaiman et al., 2015) are mapped into the same MATPOWER load flow program in order to evaluate the total transmission loss of the system. Table 4.5 presents the optimal results of the control variables after optimization. It is worth to note that all the values of the control variables optimized by MFO, ALO, DE and GA are within the range of their limits as constrained. Table 4.6 illustrates the comparison of the minimum power loss obtained by different optimization algorithms. From Table 4.5 and Table 4.6, it is obvious that the proposed MFO performs better than other selected algorithms by obtaining the lowest minimum power loss. Furthermore, two best results of MFO by using 20 search agents and 30 search agents are also tabulated in Table 4.5. Both MFO#20 and MFO#30 indicate implementing MFO with 20 search agents and 30 search agents, respectively. It can be observed that the results obtained using 30 search agents is slightly better than 20 search agents by producing lower power loss.

In order to assess the superiority of MFO algorithm, its simulation results are compared with the best well-known optimization techniques. These included PSO, DE and GA. For PSO, the optimal results of the control variables are obtained from (Khazali & Kalantar, 2011) and mapped into the MATPOWER load flow program to calculate the transmission power loss. However, for DE and GA, the optimal results for both the control variables and transmission power loss are obtained by running the MATLAB program. All the initial settings as well as operating constraints are set according to the settings utilized by proposed MFO. This is to make sure reasonable comparison can be made between MFO and the selected well-known algorithms. The comparison between MFO and the selected algorithms (PSO, DE and GA) gives about 22.02 %, 6.17 % and 5.97 % of improvement in total power loss reduction. This achievement shows that the proposed MFO is able to improve the reduction of total power loss and leads to a better results than the best well-known algorithms in solving ORPD. Table 4.8 summarizes the percentage of loss reduction improved by MFO using 30 search agents compared against other selected techniques. The percentages of improvement in power loss reduction are calculated by the formula below:

$$Loss\ reduction\ improved\ by\ MFO(\%) = \frac{P_{Loss}\ of\ reviewed\ method - P_{Loss}\ of\ MFO}{P_{Loss}\ of\ reviewed\ method} \times 100\% \qquad 4.2$$

Table 4.7 presents the comparison of the percentage of loss reduction for different optimization algorithms. The transmission power loss for the base case is

5.663 MW. The percentage of loss reduction is calculated by utilizing the equation below:

$$Loss\ reduction\,(\%) = \frac{base\ case\ loss - best\ case\ loss}{base\ case\ loss} \times 100\% \qquad 4.3$$

Based on Table 4.7, it can be noticed that MFO algorithm obtained the lowest minimum transmission power loss and the highest percentage of loss reduction. The proposed MFO is able to reduce about 19.01 % of loss reduction from the base case loss of 5.663 MW. The minimum power loss obtained by MFO is 4.5865 MW (best case loss). On the other hand, the recently best result obtained from literature which is optimized using GWO (Mohd Herwan Sulaiman et al., 2015). The total power loss is reduced by GWO to a minimum value of 4.5984 MW from the base case loss. It is about 18.80 % of loss reduction. Thus, MFO is able to improve 0.26 % reduction of power loss as compared with GWO. From this achievement, it is proved that MFO is able to excel GWO algorithm in solving ORPD.

Furthermore, the simulation results of MFO for minimum power loss, percentage of loss reduction and percentage of improvement in loss reduction are also compared with other recently proposed algorithms as presented in Table 4.7 and Table 4.8. Undeniably, MFO is also validated to be able to excel hybrid MICA-IWO. It gives about 5.35 % improvement in reduction of total power loss as compare with the hybrid algorithm. Judging from the tabulated results in the tables, it is concluded that the proposed MFO gives the best performance on solving ORPD by obtaining the lowest transmission power loss and highest percentage of loss reduction. This again proved that MFO is robust among all other rival approaches.

Table 4.5 Optimal results of the control variables after optimization for test case 1, IEEE 30-bus system

Control Variables	Initial (Base Case)	SGA	PSO	HSA	ICA	IWO	MICA-IWO
V_1	1.0600	1.0512	1.0313	1.0726	1.0785	1.0697	1.0797
V_2	1.0450	1.0421	1.0114	1.0625	1.0694	1.0604	1.0706
V_5	1.0100	1.0322	1.0221	1.0399	1.0470	1.0369	1.0484
V_8	1.0100	0.9815	1.0031	1.0422	1.0471	1.0386	1.0487
V_{11}	1.0820	0.9766	0.9744	1.0318	1.0349	1.0297	1.0752
V_{13}	1.0710	1.1000	0.9987	1.0681	1.0711	1.0557	1.0707
T_{6-9}	1.0780	0.9500	0.9700	1.0100	1.0800	1.0500	1.0300
T_{6-10}	1.0690	0.9800	1.0200	1.0000	0.9500	0.9600	0.9900
T_{4-12}	1.0320	1.0400	1.0100	0.9900	1.0000	0.9700	1.0000
T_{27-28}	1.0680	1.0200	0.9900	0.9700	0.9700	0.9700	0.9800
Q_{C3}	1.0000	12.000	17.000	34.000	−6.000	8.0000	−7.000
Q_{C10}	19.000	−10.00	13.000	12.000	36.000	35.000	23.000
Q_{C24}	4.3000	30.000	23.000	10.000	11.000	11.000	12.000
P_{Loss} (MW)	5.6630	6.5318	5.8815	5.1091	4.8489	4.9205	4.8458

Table 4.5 Continued

Control Variables	GWO	DE	GA	ALO	MFO#20	MFO#30
					20 Search Agents	30 Search Agents
V_1	1.1000	1.0953	1.0721	1.1000	1.1000	1.1000
V_2	1.0962	1.0860	1.0630	1.0948	1.0943	1.0946
V_5	1.0800	1.0626	1.0377	1.0759	1.0752	1.0756
V_8	1.0804	1.0651	1.0445	1.0774	1.0770	1.0772
V_{11}	1.0935	1.0266	1.0132	1.0761	1.0696	1.0868
V_{13}	1.1000	1.0143	1.0898	1.1000	1.1000	1.1000
T_{6-9}	1.0400	1.0178	1.0221	1.0300	1.0500	1.0411
T_{6-10}	0.9500	0.9793	0.9917	1.0000	0.9500	0.9501
T_{4-12}	0.9500	0.9778	0.9964	1.0100	0.9549	0.9554
T_{27-28}	0.9500	1.0089	0.9710	0.9800	0.9578	0.9575
Q_{C3}	12.000	20.224	5.3502	-1.000	7.0538	7.1032
Q_{C10}	30.000	9.5843	36.000	25.000	36.000	30.796
Q_{C24}	8.0000	13.030	12.418	11.000	9.8889	9.8981
P_{Loss} (MW)	4.5984	4.8881	4.8775	4.6161	4.5867	4.5865

Table 4.6 Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 1, IEEE 30-bus system)

Optimization Algorithms	Minimum Power Loss (MW)
Simple Genetic Algorithm, SGA (Khazali & Kalantar, 2011)	6.5318
Particle Swarm Optimization, PSO (Khazali & Kalantar, 2011)	5.8815
Harmony Search Algorithm, HSA (Khazali & Kalantar, 2011)	5.1091
Imperialist Competitive Algorithm, ICA (M. Ghasemi et al., 2014)	4.8489
Invasive Weed Optimization, IWO (M. Ghasemi et al., 2014)	4.9205
Modified Imperialist Competitive Algorithm and Invasive Weed Optimization, MICA-IWO (M. Ghasemi et al., 2014)	4.8458
Grey Wolf Optimizer, GWO (Mohd Herwan Sulaiman et al., 2015)	4.5984
Differential Evolution, DE	4.8881
Genetic Algorithm, GA	4.8775
Ant Lion Optimizer, ALO	4.6161
MFO#20	4.5867
MFO#30	4.5865

Table 4.7 Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 1, IEEE 30-bus system)

Compared Items	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction (%)
Base Case	5.6630	N/A
HSA	5.1091	9.78
MICA-IWO	4.8458	14.43
GWO	4.5984	18.80
DE	4.8881	13.68
GA	4.8775	13.87
ALO	4.6161	18.49
MFO#20	4.5867	19.01
MFO#30	4.5865	19.01

Table 4.8 The percentage of loss reduction improved by MFO#30 compared with other optimization algorithms (test case 1, IEEE 30-bus system)

Optimization Algorithms	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction Improved by MFO (%)
MFO#30	4.5865	N/A
HSA	5.1091	10.23
MICA-IWO	4.8458	5.35
PSO	5.8815	22.02
DE	4.8881	6.17
GA	4.8775	5.97
GWO	4.5984	0.26
ALO	4.6161	0.64

The convergence performances of proposed MFO are illustrated in Figure 4.1 with four different numbers of search agents. The graph is plotted in term of power loss (MW) versus iteration (maximum iterations=150). For this graph, it is important to emphasize that the results plotted are the best results that have been chosen from 30 free running simulations. It can be noted that the system power loss optimized using different number of search agents is reduced over the course of iteration and converge to a minimum value. For more details, the minimum power loss obtained by 10, 20, 30 and 40 search agents are 4.7510 MW, 4.5867 MW, 4.5864 MW and 4.5866 MW, respectively. Thus, 30 search agents can get the lowest minimum power loss among other number of search agents. It also can be observed from the graph that 30 search agents are able to get merely better convergence characteristics than 20 search agents and 40 search agents. In the nutshell, it can be concluded that a better convergence characteristic can be attained by increasing the number of the search agents. However, 30 search agents perform the best convergence characteristics among others including 40 search agents. Therefore, 30 search agents are taken into account for all the test cases throughout this research.

Figure 4.2 and Figure 4.3 illustrate the performance characteristics by implementing MFO algorithm with 20 search agents and 30 search agents, respectively for 30 free running simulations. For 20 search agents, the best, average and worst results of power losses are 4.5867 MW, 4.6749 MW and 5.2313 MW, respectively. On the other hand, the best, average and worst results of losses using 30 search agents are 4.5864 MW, 4.6081 MW and 4.7486 MW, respectively. The comparison of the results of power losses between 20 search agents and 30 search agents is summarized in Table

4.9. It can be noticed that 30 search agents are able to obtain better solutions in terms of best, average and worst results than 20 search agents. Furthermore, Figure 4.2 and Figure 4.3 are plotted in a same graph as shown in Figure 4.4 in order to further compare their performances. Based on Figure 4.4, it is worth to emphasize that 30 search agents can obtain a consistent results with a smaller range of power loss as compared against 20 search agents. The range of power losses utilizing 30 search agents is in an interval of [4.58 MW, 4.76 MW] while the range of power losses using 20 search agents is in an interval of [4.5 MW, 5.3 MW]. Therefore, it again shows that MFO algorithm can attain better results by using 30 search agents.

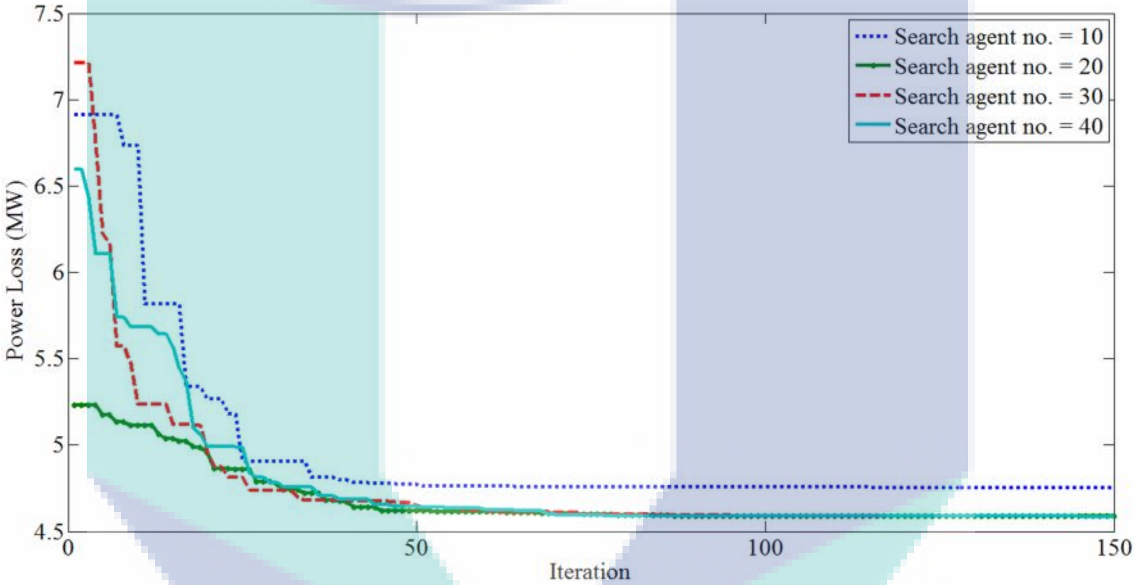

Figure 4.1 Convergence performances of power loss of IEEE 30-bus system for different numbers of search agents using MFO algorithm

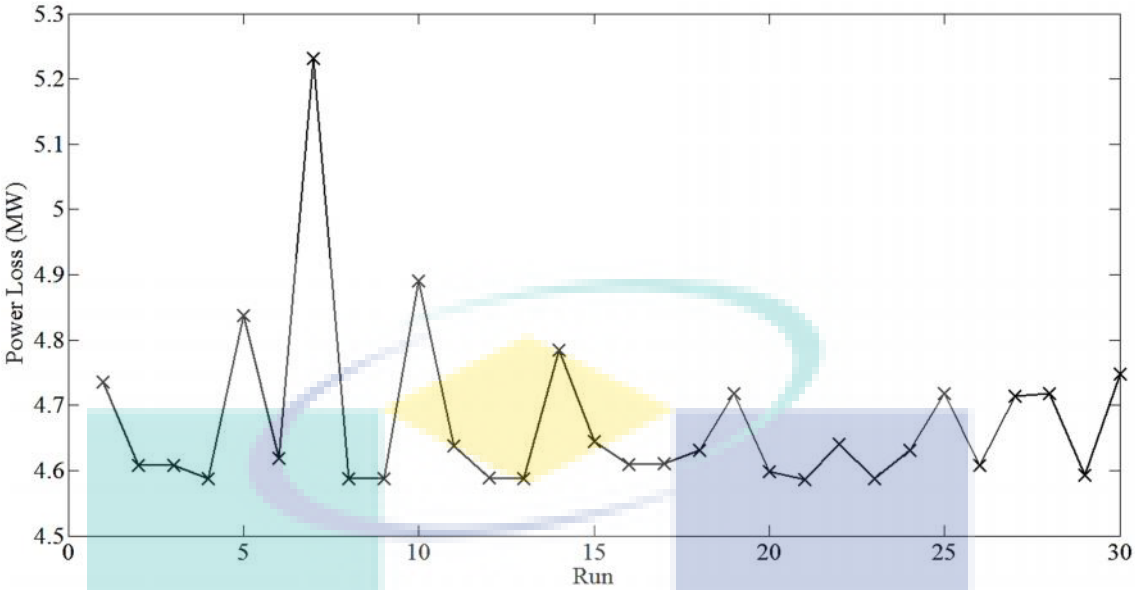

Figure 4.2 Performance characteristics of MFO using 20 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)

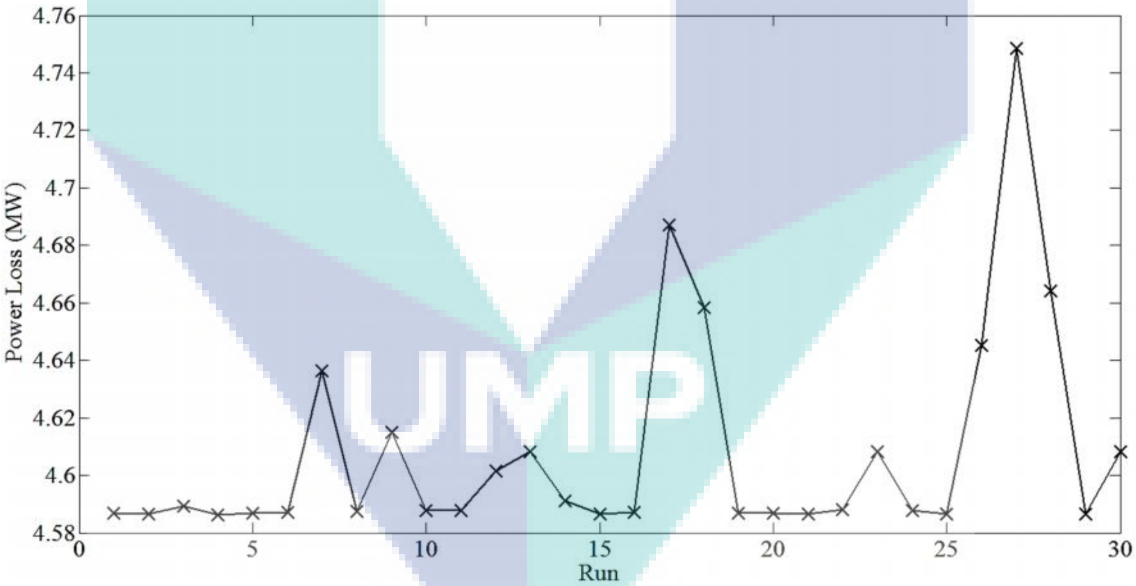

Figure 4.3 Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)

Table 4.9 Comparison of performances between 20 search agents and 30 search agents

Compared Items (P_{Loss})	20 Search Agents	30 Search Agents
Best Result (MW)	4.5867	4.5864
Average Result (MW)	4.6749	4.6081
Worst Result (MW)	5.2313	4.7486

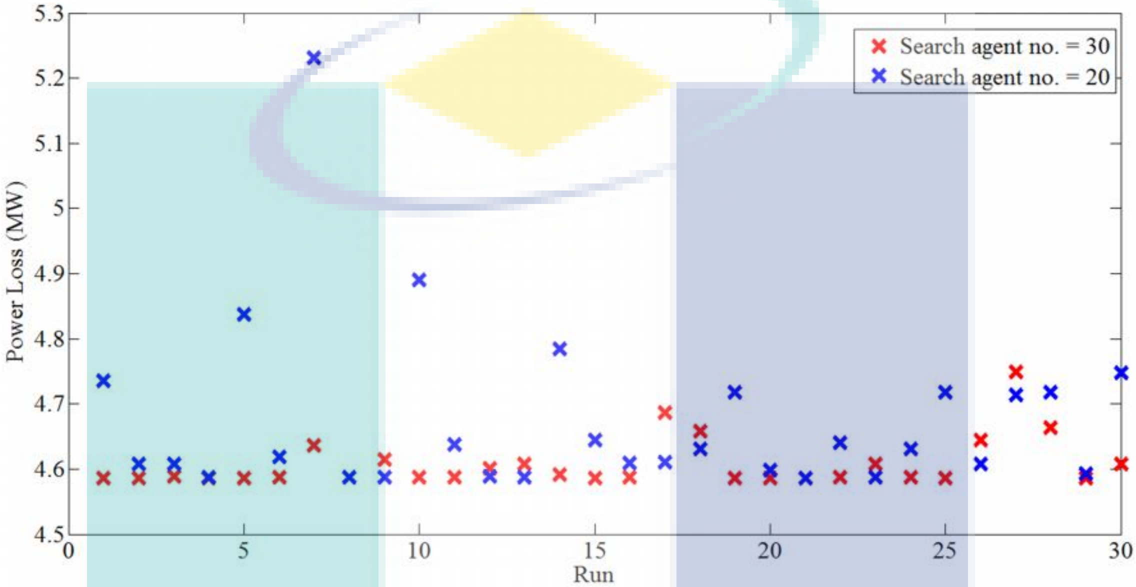

Figure 4.4 Comparison of performances between 20 and 30 search agents for 30 free running simulations (test case 1, IEEE 30-bus system)

The optimized solutions of voltage deviation as second objective function of ORPD are tabulated in Table 4.10. It is worth to highlight that the solutions for the best, average and worst results of MFO and ALO are obtained from 30 free running simulations. Their results are compared with the results obtained by SGA, PSO, HSA and GWO. Judging from the simulation results, it is proven that MFO is able to produce the smallest voltage deviation among other rival approaches. The best deviation of MFO is 0.1215 p.u. Additionally, MFO also produced the lowest average and worst deviations as compared to others. The average and worst deviations of MFO are 0.1330 p.u and 0.1561 p.u, respectively. On the other hand, ALO is able to obtain the best and average deviations at load buses compared to others except MFO. However, ALO also produces the highest worst deviation among others.

Figure 4.5 illustrates the comparison of performance characteristics for the voltage deviations optimized by ALO and MFO, respectively. By referring to the graph, it is obvious that MFO algorithm can produce a consistent voltage deviation result throughout 30 simulations. Moreover, the range of the voltage deviation optimized using MFO is within 0.12 p.u and 0.15 p.u while the range of the voltage deviation produced by ALO is within 0.12 p.u and 0.18 p.u. This comparison again concluded that MFO is able to produce a smaller range of voltage deviation compared to ALO and other selected approaches as discussed above.

Table 4.10 Comparison of voltage deviation for different optimization algorithms (test case 1, IEEE 30-bus system)

Compared Items	SGA	PSO	HSA	GWO (Mohd Herwan Sulaiman et al., 2015)	ALO	MFO
	(Khazali & Kalantar, 2011)					
Best Deviation (p.u)	0.1501	0.1424	0.1349	0.1260	0.1246	0.1215
Average Deviation (p.u)	0.1523	0.1496	0.1443	0.1448	0.1416	0.1330
Worst Deviation (p.u)	0.1717	0.1639	0.1589	0.1727	0.1815	0.1561

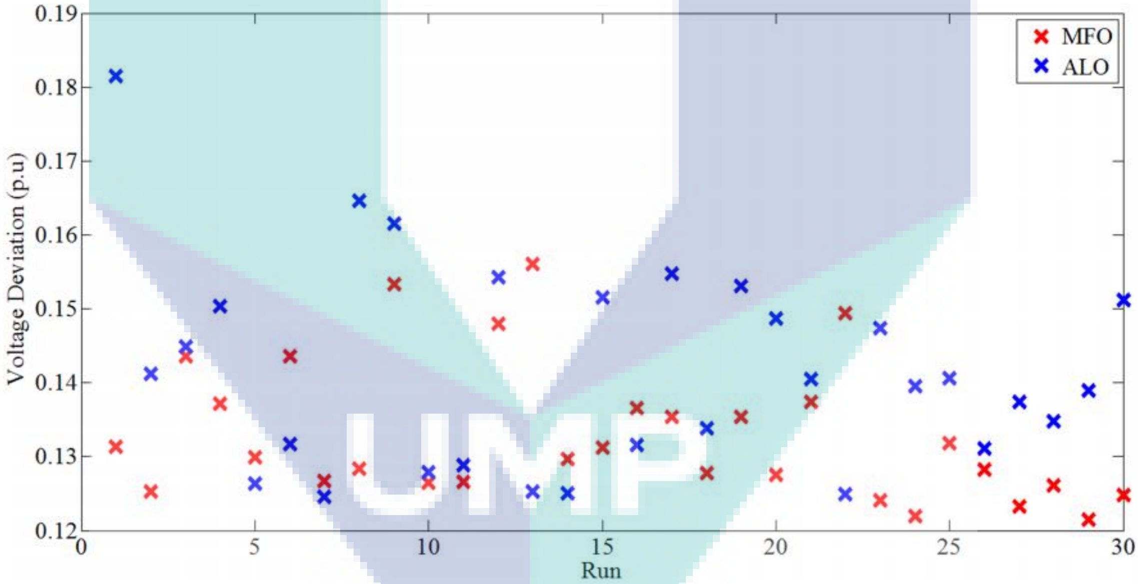

Figure 4.5 Comparison of voltage deviation performances between ALO and MFO for 30 free running simulations (test case 1, IEEE 30-bus system)

4.3 Test Case 2: IEEE 30-Bus System with 19 Control Variables

For the second case study, the same IEEE 30-bus system will be used. The initial setting (base case) of the control variables are set according to the values as tabulated in Table 4.12. The initial transmission loss of this case study is found to be

5.674 MW. In this test case, a total of 19 control variables are considered to be optimized. These included six generator buses voltage, four transformers and nine compensator elements. The four transformers are located at lines 6-9, 6-10, 4-12 and 27-28 while the reactive compensators are located at buses 10, 12, 15, 17, 20, 21, 23, 24 and 29. Table 4.11 presents the boundaries setting for the control variables used in this case study. In this case study, the limit of the generator buses voltage and sizing of compensators are decreased to a smaller range as compared to case test 1. Moreover, both of the real and reactive load demands for this test case are same as test case 1.

Table 4.12 tabulates the optimization results obtained using MFO and DE. In this context, it is vital to highlight that the minimum power loss (P_{Loss}=4.5179 MW) tabulated in Table 4.12 is different with the minimum power loss (P_{Loss}=4.5550 MW) reported in (El Ela et al., 2011) for the same constraints setting. The power losses report in this table are calculated through the same MATPOWER load flow program. The optimal results of control variables obtained by DE are extracted from (El Ela et al., 2011). They are mapped into the same MATPOWER load flow program used in this test case. This is to ensure fair and reasonable comparison can be made between DE and proposed MFO. In addition, it is worth to emphasize that the number of populations and maximum generations used by El Ela et al. to obtain the optimal results of control variables are 150 and 500, respectively. On the other hand, MFO used only 30 search agents and 150 maximum iterations to reach the optimal solutions.

Judging from Table 4.12, it is undeniable that the minimum power loss optimized by proposed MFO (P_{Loss}=4.5128 MW) is lower than the loss obtained by DE (P_{Loss}=4.5179 MW). By comparing MFO to DE method, it is about 0.11 % reduction of total loss (calculated using Eq. 4.2). Furthermore, MFO is also able to reduce about 20.47 % of loss from the base case loss while DE reduces about 20.38 % of loss from the base case loss. The difference in percentages of loss reduction between DE and MFO are relatively small which is only 0.09%. In addition, it is important to note that all the optimized results of control variables are converged within their respective limit ranges as constrained in Table 4.11.

For the second objective function of ORPD, the optimal values of the control variable settings obtained by proposed MFO are tabulated in the sixth column of Table 4.12. Obviously, the voltage deviation is reduced from 1.1606 p.u (base case loss) to

0.0897 p.u with a total reduction of 92.27 %. The comparison with the results optimized by DE as reported in (El Ela et al., 2011) are given in the same table as well. From the comparison, the proposed MFO produces a better solution over DE which again proved the effectiveness of MFO on minimizing the voltage deviation.

Table 4.11 Setting of variables limit for test case 2, IEEE 30-bus system

Control Variables	Lower Bound	Upper Bound
Generator Buses Voltage	0.95 p.u	1.10 p.u
Transformers Tap Setting	0.90 p.u	1.10 p.u
Reactive Compensators Sizing	0 MVar	5 MVar

Table 4.12 Comparison of the proposed MFO with the results obtained by DE

Control Variables	Initial (Base Case)	Minimization of P_{Loss}		Minimization of Voltage Deviation	
		DE	MFO	DE	MFO
V_1	1.0500	1.1000	1.1000	1.1000	0.9872
V_2	1.0400	1.0931	1.0943	0.9918	0.9996
V_5	1.0100	1.0736	1.0747	1.0179	1.0207
V_8	1.0100	1.0756	1.0766	1.0183	1.0123
V_{11}	1.0500	1.1000	1.1000	1.0114	0.9713
V_{13}	1.0500	1.1000	1.1000	1.0282	1.0574
$T_{6\text{-}9}$	1.0780	1.0465	1.0433	1.0265	0.9840
$T_{6\text{-}10}$	1.0690	0.9097	0.9000	0.9038	0.9022
$T_{4\text{-}12}$	1.0320	0.9867	0.9791	1.0114	1.0734
$T_{27\text{-}28}$	1.0680	0.9689	0.9647	0.9635	0.9607
Q_{C10}	1.0000	5.0000	5.0000	4.9420	4.5240
Q_{C12}	1.0000	5.0000	5.0000	1.0885	0.4746
Q_{C15}	1.0000	5.0000	4.8055	4.9855	5.0000
Q_{C17}	1.0000	5.0000	5.0000	0.2393	0.8515
Q_{C20}	1.0000	4.4060	4.0263	4.9958	5.0000
Q_{C21}	1.0000	5.0000	5.0000	4.9075	4.9965
Q_{C23}	1.0000	2.8000	2.5193	4.9863	5.0000
Q_{C24}	1.0000	5.0000	5.0000	4.9663	4.9843
Q_{C29}	1.0000	2.5979	2.1925	2.2325	1.7019
P_{Loss} (MW)	5.6740	4.5179	4.5128	N/A	N/A
Voltage Deviation (p.u)	1.1606	N/A	N/A	0.0911	0.0897
Percentage of Loss Reduction (%)	N/A	20.38	20.47	N/A	N/A
Percentage of Deviation Reduction (%)	N/A	N/A	N/A	92.15	92.27

4.4 Test Case 3: IEEE 30-Bus System with 25 Control Variables

For the third case study, the system demand for real and reactive loads are the same as test case 1. There are additional numbers of control variables that need to be optimized with a total of 25 control variables are considered. The boundaries setting for this case study are summarized in Table 4.13 along with the initial values for the control parameters. By referring to the table, all the values of the control variables are presented in terms of per unit (p.u). In addition, there are four transformers and nine compensators to be optimized in this test case. The transformers are situated at lines 6-9, 6-10, 4-12 and 27-28 while the compensators are placed at buses 10, 12, 15, 17, 20, 21, 23, 24 and 29.

The optimal results of control variables and power loss are presented in Table 4.14 while Table 4.15 depicts the comparison of the optimal transmission loss between MFO and other algorithms. For optimization purposes, the simulation parameters for MFO and ALO including number of search agents and maximum iterations are set as 30 and 150, respectively. According to Table 4.14, all the results of GWO are extracted directly from (Mohd Herwan Sulaiman et al., 2015). On the other hand, the optimal solutions of control variables obtained by ABC (Ayan & Kılıç, 2012) are mapped into the MATPOWER load flow program to calculate the total transmission loss. Again, this is to assure a fair and reasonable comparison can be made.

As compared in Table 4.15, proposed MFO is proven to obtain best minimized transmission power loss (P_{Loss}=2.8298 MW). It is also validated to be able to reduce the highest percentage of loss among other reviewed algorithms. It is about 50.76 % of loss reduction which can yield a big impact on solving ORPD problems in power system. The comparison in percentage of loss reduction by different optimization algorithms are tabulated in Table 4.16. Based on the simulation results, it is undeniable that MFO is superior compared to ALO, MVO, GWO and ABC. It gives about 2.55 %, 3.46 %, 3.67 % and 6.96 %, respectively (calculated using Eq. 4.2) in improving the reduction of total system loss (as presented in Table 4.17).

Table 4.13 The lower and upper limits of the control variables along with the initial settings for test case 3, IEEE 30-bus system

Control Variables	Boundaries		Initial (Base Case)
	Lower	Upper	
P_1 (p.u)	0.50	2.00	2.6020
P_2 (p.u)	0.20	0.80	0.8000
P_5 (p.u)	0.15	0.50	0.5000
P_8 (p.u)	0.10	0.35	0.2000
P_{11} (p.u)	0.10	0.30	0.2000
P_{13} (p.u)	0.12	0.40	0.2000
V_1 (p.u)	1.00	1.10	1.0600
V_2 (p.u)	1.00	1.10	1.0450
V_5 (p.u)	1.00	1.10	1.0100
V_8 (p.u)	1.00	1.10	1.0100
V_{11} (p.u)	1.00	1.10	1.0820
V_{13} (p.u)	1.00	1.10	1.0710
T_{6-9} (p.u)	0.90	1.10	1.0780
T_{6-10} (p.u)	0.90	1.10	1.0690
T_{4-12} (p.u)	0.90	1.10	1.0320
T_{27-28} (p.u)	0.90	1.10	1.0680
Q_{C10} (p.u)	0	0.05	0.0100
Q_{C12} (p.u)	0	0.05	0.0100
Q_{C15} (p.u)	0	0.05	0.0100
Q_{C17} (p.u)	0	0.05	0.0100
Q_{C20} (p.u)	0	0.05	0.0100
Q_{C21} (p.u)	0	0.05	0.0100
Q_{C23} (p.u)	0	0.05	0.0100
Q_{C24} (p.u)	0	0.05	0.0100
Q_{C29} (p.u)	0	0.05	0.0100
P_{Loss} (MW)			5.7470

Table 4.14 Optimal results of the control variables after optimization for test case 3, IEEE 30-bus system

Control Variables	ABC	GWO	MVO	ALO	MFO
P_1	0.5462	0.5161	0.5266	2.0000	0.5000
P_2	0.7863	0.7979	0.7990	0.8000	0.8000
P_5	0.4903	0.5000	0.4883	0.5000	0.5000
P_8	0.3477	0.3493	0.3497	0.3500	0.3500
P_{11}	0.2999	0.3000	0.3000	0.3000	0.3000
P_{13}	0.3945	0.4000	0.3997	0.4000	0.4000
V_1	1.0927	1.1000	1.1000	1.1000	1.1000
V_2	1.0880	1.0981	1.0988	1.0992	1.0977
V_5	1.0695	1.0766	1.0833	1.0819	1.0798
V_8	1.0722	1.0870	1.0890	1.0906	1.0873
V_{11}	1.0860	1.0970	1.0999	1.0985	1.0995
V_{13}	1.0926	1.1000	1.1000	1.0991	1.0998
$T_{6\text{-}9}$	0.9983	0.9912	1.0446	0.9932	1.0641
$T_{6\text{-}10}$	0.9994	1.0402	0.9568	1.0854	0.9000
$T_{4\text{-}12}$	0.9984	1.0332	0.9889	1.0224	0.9884
$T_{27\text{-}28}$	1.0034	0.9913	0.9866	1.0087	0.9726
Q_{C10}	0.0155	0.0436	0.0474	0.0497	0.0500
Q_{C12}	0.0394	0.0103	0.0305	0.0425	0.0500
Q_{C15}	0.0347	0.0268	0.0273	0.0311	0.0500
Q_{C17}	0.0331	0.0500	0.0365	0.0500	0.0500
Q_{C20}	0.0332	0.0006	0.0450	0.0388	0.0398
Q_{C21}	0.0395	0.0300	0.0483	0.0500	0.0500
Q_{C23}	0.0130	0.0057	0.0410	0.0500	0.0193
Q_{C24}	0.0371	0.0459	0.0400	0.0500	0.0498
Q_{C29}	0.0399	0.0044	0.0474	0.0467	0.0197
P_{Loss} (MW)	3.0415	2.9377	2.9311	2.9039	2.8298

Table 4.15 Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 3, IEEE 30-bus system)

Optimization Algorithms	Minimum Power Loss (MW)
Artificial Bee Colony, ABC (Ayan & Kılıç, 2012)	3.0415
Grey Wolf Optimizer, GWO (Mohd Herwan Sulaiman et al., 2015)	2.9377
Multi-Verse Optimizer, MVO (Mohd Herwan Sulaiman et al., 2016)	2.9311
Ant Lion Optimizer, ALO	2.9039
Moth-Flame Optimization, MFO	2.8298

Table 4.16 Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 3, IEEE 30-bus system)

Compared Items	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction (%)
Base Case	5.7470	N/A
ABC	3.0415	47.08
GWO	2.9377	48.88
MVO	2.9311	49.00
ALO	2.9039	49.47
MFO	2.8298	50.76

Table 4.17 The percentage of loss reduction improved by MFO compared with other optimization algorithms (test case 3, IEEE 30-bus system)

Optimization Algorithms	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction Improved by MFO (%)
MFO	2.8298	N/A
ABC	3.0415	6.96
GWO	2.9377	3.67
MVO	2.9311	3.46
ALO	2.9039	2.55

The performance in terms of best, average and worst results of proposed MFO is presented in Table 4.18. Its results are compared with the results obtained by ALO in this research. Again, it is vital to mention that the best, average and worst results for both ALO and MFO are obtained from 30 free running simulations. In addition, Figure 4.6 depicts the performance characteristics of MFO undergoes 30 free running simulations. The best, average and worst solutions of MFO are 2.8298 MW, 2.8909 MW and 3.1273 MW, respectively. By referring to the simulation results tabulated in Table 4.18, it is noticed that MFO is able to produce the lower solutions in terms of best, average and worst results than ALO. Figure 4.7 illustrates the comparison of performance characteristics between MFO and ALO in terms of minimizing the system power loss for 30 simulations. From the graph, it is again validated that MFO can produce the lower range of power loss compared to ALO. Most of the solutions obtained by MFO are situated in the interval of [2.8 MW, 3.0 MW] while the results of ALO are mostly located in the interval of [2.9 MW, 3.1 MW]. For this case study, the second objective function of ORPD is not conducted as this test case involving the optimal real power output for each generator, P_G.

Table 4.18 Comparison of power loss between ALO and MFO (test case 3, IEEE 30-bus system)

Compared Item (P_{Loss})	ALO	MFO
Best Result (MW)	2.9039	2.8298
Average Result (MW)	2.9937	2.8909
Worst Result (MW)	3.1280	3.1273

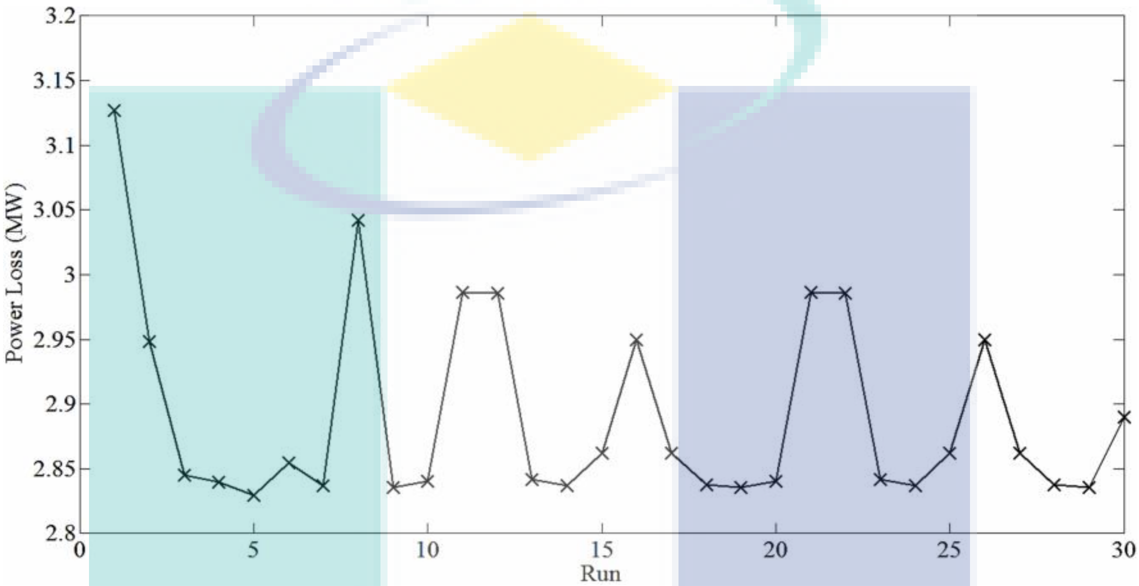

Figure 4.6 Performance characteristics of MFO for 30 free running simulations (test case 3, IEEE 30-bus system)

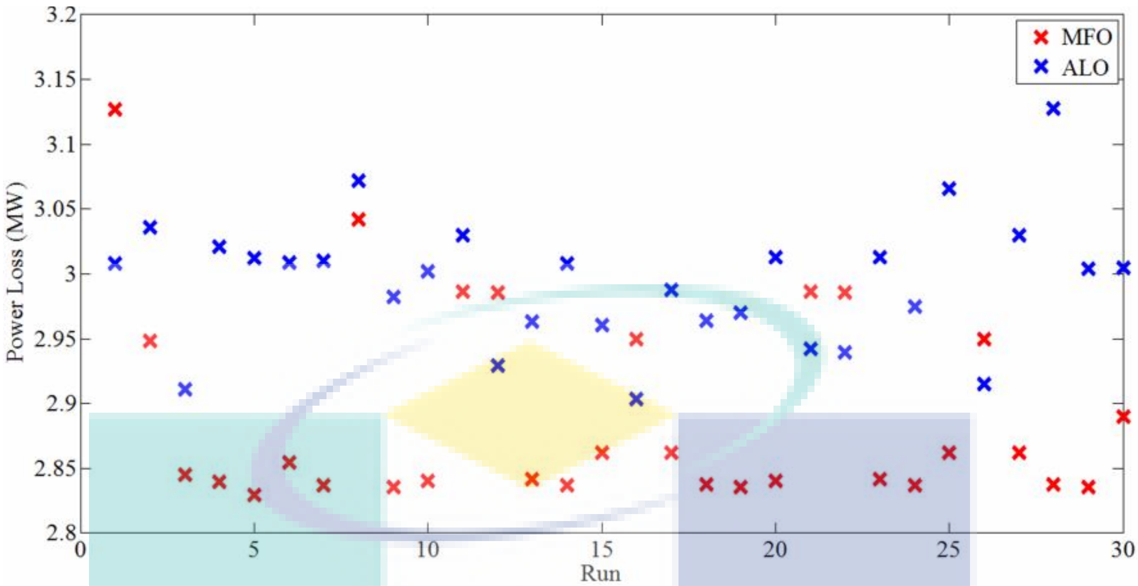

Figure 4.7 Comparison of power loss performances between ALO and MFO for 30 free running simulations (test case 3, IEEE 30-bus system)

4.5 Test Case 4: IEEE 57-Bus System with 25 Control Variables

In order to further validate the effectiveness of proposed MFO on solving ORPD problem, a medium test case of IEEE 57-bus system is used. For this test system, a total of 25 numbers of control variables is considered to be optimized; which consists of seven generators, 15 tap changers and three reactive compensators. The voltages of generator buses have been constrained within limits between 0.94 p.u and 1.06 p.u. The operating range of all the transformers and compensators are constrained by their minimum and maximum limits as presented in Table 4.19. The load demands for IEEE 57-bus system is set as

$$S = P + jQ = 12.508 + j3.364 p.u \qquad 4.4$$

where the total system load is 1250.8 MW and the initial loss is 27.864 MW (base case loss). For optimization purposes, the settings of the simulation parameters for MFO, ALO, DE and GA are same as the values presented in Table 4.4 except for the maximum number of iterations. The maximum iterations for this test case is set as 300 and the NFE to reach the best optimal solutions is 9000.

Table 4.19 Setting of variables limits for test case 4, 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 4.20 and Table 4.21 tabulate the optimal results of control variables and minimum power loss obtained for this test system by different optimization algorithms. For fair comparison, the same load flow program is used by the other reviewed optimization algorithms (Dai, Chen, Zhu, & Zhang, 2009; M H Sulaiman et al., 2014; Mohd Herwan Sulaiman, Mustaffa, Daniyal, Mohamed, & Aliman, 2006) throughout this case study. The results for the optimal setting of control variables obtained by proposed MFO are all converged within their constrained limit ranges. Based on Table 4.21, it is proven that MFO outperforms other algorithms by producing the lowest minimum power loss (P_{Loss}=24.2529 MW). By comparing the minimum power loss of MFO with the well-known methods (PSO, DE and GA), it gives about 0.53 %, 6.56 % and 1.40 % reduction of total power loss, respectively. This improvement proved that MFO able to yield a better results with a smaller total power loss than the renowned algorithms on solving ORPD problem. Table 4.23 tabulates the percentage of loss reduction improved by proposed MFO compared to other reviewed algorithms. The percentages are calculated using Eq. 4.2.

Table 4.22 illustrates the comparison of power loss reduction before and after optimization by various algorithms. It is undeniable that proposed MFO is superior than other algorithms by minimizing the total transmission loss to 24.2529 MW from the base case loss of 27.864 MW. It reduces about 12.96 % of power loss which is considered as the highest percentage of loss reduction against other reviewed algorithms. Furthermore, based on the comparison between MFO and recently developed algorithms (GWO and ALO), it gives about 2.02 % and 2.06 % reduction of total power loss, respectively. This achievement means that MFO is able to improve around 2 % of loss reduction from these two methods. On the other hand, the latest best results obtained from literatures are optimized by using CSA (M H Sulaiman et al., 2014) and SOA (Dai et al., 2009). They produce about 12.93 % and 12.91 % loss reduction from the base case loss, respectively. Nevertheless, MFO still excels their results. By comparing MFO to CSA and SOA, it gives about 0.06 % and 0.04 % improvement in reduction of total power loss. In a nutshell, it is concluded that proposed MFO is outperformed among these rival algorithms for solving ORPD problems on a medium test system.

Table 4.20 Optimal results of the control variables after optimization for test case 4, IEEE 57-bus system

Control Variables	Initial (Base Case)	GSA	PSO	CSA	FA	GWO	SOA
V_1	1.0400	1.0600	1.0600	1.0600	1.0600	1.0600	1.0600
V_2	1.0100	1.0582	1.0600	1.0582	1.0572	1.0562	1.0580
V_3	0.9850	1.0462	1.0483	1.0466	1.0428	1.0370	1.0437
V_6	0.9800	1.0391	1.0423	1.0409	1.0366	1.0202	1.0352
V_8	1.0050	1.0600	1.0600	1.0587	1.0541	1.0449	1.0548
V_9	0.9800	1.0432	1.0432	1.0417	1.0355	1.0294	1.0369
V_{12}	1.0150	1.0379	1.0387	1.0377	1.0320	1.0319	1.0336
$T_{4–18}$	0.9700	0.9054	0.9000	0.9440	0.9312	0.9847	1.0000
$T_{4–18}$	0.9780	0.9978	1.1000	1.0182	0.9901	0.9326	0.9600
$T_{21–20}$	1.0430	1.0021	1.0314	1.0207	0.9845	0.9576	1.0100
$T_{24–26}$	1.0430	1.0180	1.0097	1.0110	1.0112	0.9968	1.0100
$T_{7–29}$	0.9670	0.9712	0.9754	0.9744	0.9683	0.9636	0.9700
$T_{34–32}$	0.9750	0.9692	0.9746	0.9721	0.9657	0.9812	0.9700
$T_{11–41}$	0.9550	0.9683	0.9000	0.9015	0.9762	1.0621	0.9000
$T_{15–45}$	0.9550	0.9717	0.9726	0.9723	0.9653	0.9755	0.9700
$T_{14–46}$	0.9000	0.9530	0.9538	0.9537	0.9524	0.9639	0.9500
$T_{10–51}$	0.9300	0.9691	0.9680	0.9664	0.9671	0.9723	0.9600
$T_{13–49}$	0.8950	0.9242	0.9264	0.9269	0.9291	0.9248	0.9200
$T_{11–43}$	0.9580	1.0387	1.1000	0.9645	1.0020	0.9554	0.9600
$T_{40–56}$	0.9580	1.0497	1.0624	0.9943	1.0224	1.1000	1.0000
$T_{39–57}$	0.9800	1.0668	1.0265	0.9737	1.0232	0.9976	0.9600
$T_{9–55}$	0.9400	0.9807	0.9764	0.9750	0.9687	0.9845	0.9700
Q_{C18}	10.000	0.1863	9.9988	9.2807	4.1934	1.8917	9.9840
Q_{C25}	5.9000	4.0488	5.9000	5.8943	4.2297	5.2489	5.9040
Q_{C53}	6.3000	4.8099	6.3000	6.2885	5.9252	5.1513	6.2880
P_{Loss} (MW)	27.8640	24.4922	24.3826	24.2619	24.4587	24.7523	24.2677

Table 4.20 Continued

Control Variables	BAT	FPA	DE	GA	ALO	MFO
V_1	1.0603	1.0599	1.0549	1.0600	1.0600	1.0600
V_2	1.0558	1.0561	1.0475	1.0556	1.0595	1.0587
V_3	1.0456	1.0472	1.0200	1.0320	1.0494	1.0469
V_6	1.0369	1.0401	1.0004	1.0187	1.0409	1.0421
V_8	1.0499	1.0585	1.0173	1.0380	1.0600	1.0600
V_9	1.0405	1.0429	1.0108	1.0257	1.0469	1.0423
V_{12}	1.0314	1.0387	1.0187	1.0258	1.0426	1.0373
T_{4-18}	0.9810	0.9834	0.9465	1.0330	1.0791	0.9501
T_{4-18}	0.9921	1.0559	1.0221	0.9056	1.0629	1.0076
T_{21-20}	1.0155	1.0308	1.0079	0.9830	1.0471	1.0063
T_{24-26}	0.9962	1.0620	0.9692	1.0028	0.9993	1.0076
T_{7-29}	0.9624	1.0342	0.9454	0.9634	0.9768	0.9752
T_{34-32}	0.9520	1.0160	1.0035	0.9835	0.9985	0.9722
T_{11-41}	0.8857	0.9482	1.0758	0.9346	0.9958	0.9000
T_{15-45}	0.9736	0.9743	0.9932	0.9669	0.9827	0.9719
T_{14-46}	0.9747	0.9478	0.9727	0.9493	0.9793	0.9536
T_{10-51}	0.9550	0.9663	0.9936	0.9632	1.0204	0.9674
T_{13-49}	0.9271	0.9448	0.9577	0.9265	0.9530	0.9279
T_{11-43}	1.0396	0.9856	1.0095	0.9605	1.0092	0.9641
T_{40-56}	1.0800	1.0715	1.0108	1.0381	1.0675	0.9998
T_{39-57}	0.9838	0.9783	1.0324	0.9815	1.0480	0.9606
T_{9-55}	1.0156	1.0379	0.9773	0.9682	1.0111	0.9790
Q_{C18}	5.6561	6.9194	3.5497	6.6369	8.8172	9.9968
Q_{C25}	2.4993	3.7103	3.4358	5.8568	5.3446	5.9000
Q_{C53}	3.0430	4.2180	1.7723	5.9162	5.4923	6.3000
P_{Loss} (MW)	24.9254	24.8419	25.9556	24.5968	24.7621	24.2529

Table 4.21 Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 4, IEEE 57-bus system)

Optimization Algorithms	Minimum Power Loss (MW)
Gravitational Search Algorithm, GSA (M H Sulaiman et al., 2014)	24.4922
Particle Swarm Optimization, PSO (M H Sulaiman et al., 2014)	24.3826
Cuckoo Search Algorithm, CSA (M H Sulaiman et al., 2014)	24.2619
Firefly Algorithm, FA (Mohd Herwan Sulaiman, Mustaffa, Daniyal, Mohamed, & Aliman, 2015)	24.4587
Grey Wolf Optimizer, GWO (Mohd Herwan Sulaiman et al., 2015)	24.7523
Seeker Optimization Algorithm, SOA (Dai et al., 2009)	24.2677
Bat Algorithm, BAT (Yang, 2010a)	24.9254
Flower Pollination Algorithm, FPA (Yang, 2012)	24.8419
Differential Evolution, DE	25.9556
Genetic Algorithm, GA	24.5968
Ant Lion Optimizer, ALO	24.7621
Moth-Flame Optimization, MFO	24.2529

Table 4.22 Comparison of percentage of loss reduction before and after optimization by MFO and other optimization algorithms (test case 4, IEEE 57-bus system)

Compared Items	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction (%)
Base Case	27.8640	N/A
GSA	24.4922	12.10
PSO	24.3826	12.49
CSA	24.2619	12.93
FA	24.4587	12.22
GWO	24.7523	11.17
SOA	24.2677	12.91
BAT	24.9254	10.55
FPA	24.8419	10.85
DE	25.9556	6.85
GA	24.5968	11.73
ALO	24.7621	11.13
MFO	24.2529	12.96

Table 4.23 The percentage of loss reduction improved by MFO compared with other optimization algorithms (test case 4, IEEE 57-bus system)

Optimization Algorithms	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction Improved by MFO (%)
MFO	24.2529	N/A
GSA	24.4922	0.98
PSO	24.3826	0.53
CSA	24.2619	0.04
FA	24.4587	0.84
GWO	24.7523	2.02
SOA	24.2677	0.06
BAT	24.9254	2.70
FPA	24.8419	2.37
DE	25.9556	6.56
GA	24.5968	1.40
ALO	24.7621	2.06

To illustrate the effectiveness of MFO on solving ORPD problems, the convergence characteristics in terms of power loss (MW) versus iteration for four different search agents are plotted in Figure 4.8. It is important to mention that 30 trail runs are taken for each of the population and the best results of each population are plotted into the graph for comparison. The maximum iterations for this test case is set as 300. From this graph, it can be clearly notice that 30 search agents produce the best and fastest convergence against others. It started to converge at 150 iterations although the maximum iteration is 300. Figure 4.9 depicts the performance characteristics of proposed MFO using 30 search agents for 30 simulations. The transmission power losses of MFO in terms of best, average and worst results are tabulated in Table 4.24. Its results are compared with the results optimized by ALO.

Based on Table 4.24, MFO is able to produce lower best and average results than ALO. However, its worst result is higher than the worst results obtained by ALO. To further illustrate the comparison between MFO and ALO, their best optimized results of the minimum power loss from 30 trail runs are plotted together in a graph as shown in Figure 4.10. The results optimized by MFO are mostly located in a range within 24 MW and 25 MW while most of the solutions minimized by ALO are situated in a range within 25 MW and 26 MW. This comparison again validated that MFO can produce a lower range of power loss than ALO throughout 30 free running simulations. Since the reviewed algorithms aforementioned does not conduct for the minimization of voltage deviation, reasonable comparison between MFO cannot be made. Therefore, the second objective function is not included in this test case.

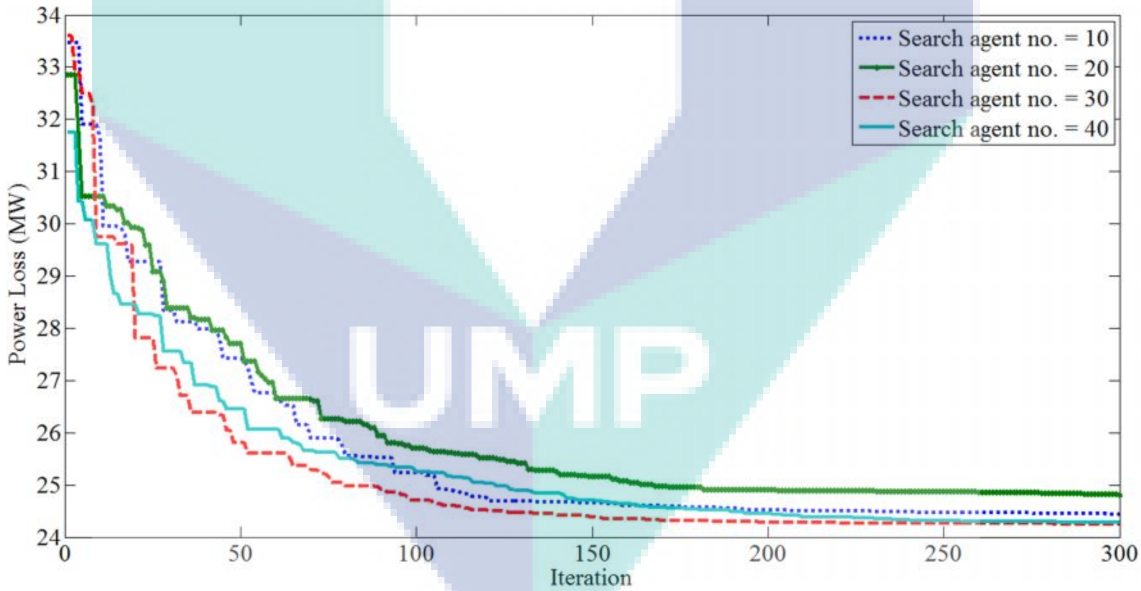

Figure 4.8 Convergence performances of power loss of IEEE 57-bus system for different numbers of search agents using MFO algorithm

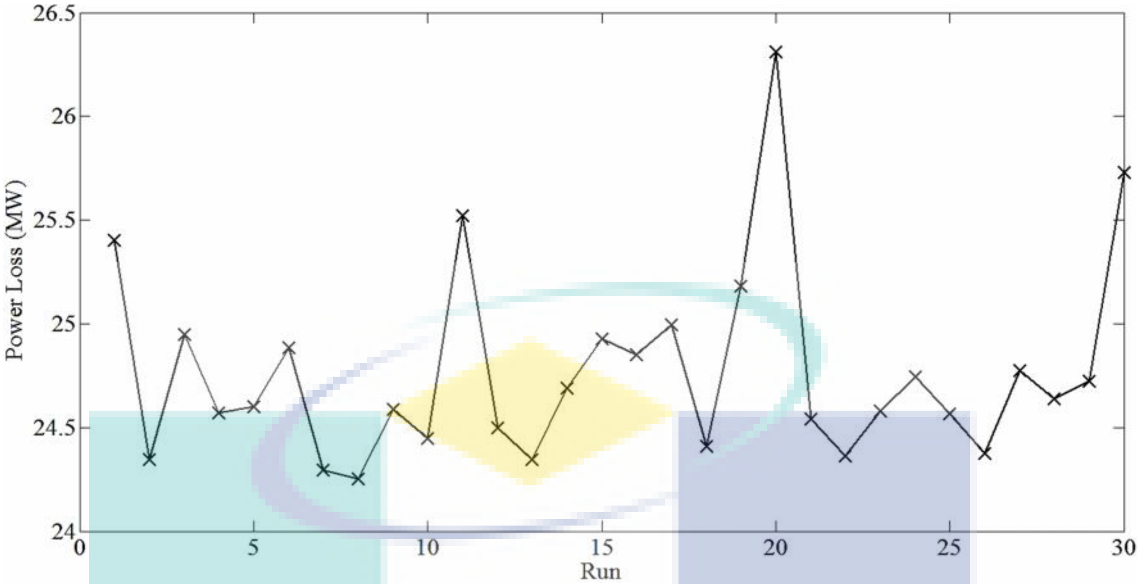

Figure 4.9 Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 4, IEEE 57-bus system)

Table 4.24 Comparison of power loss between ALO and MFO (test case 4, IEEE 57-bus system)

Compared Item (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

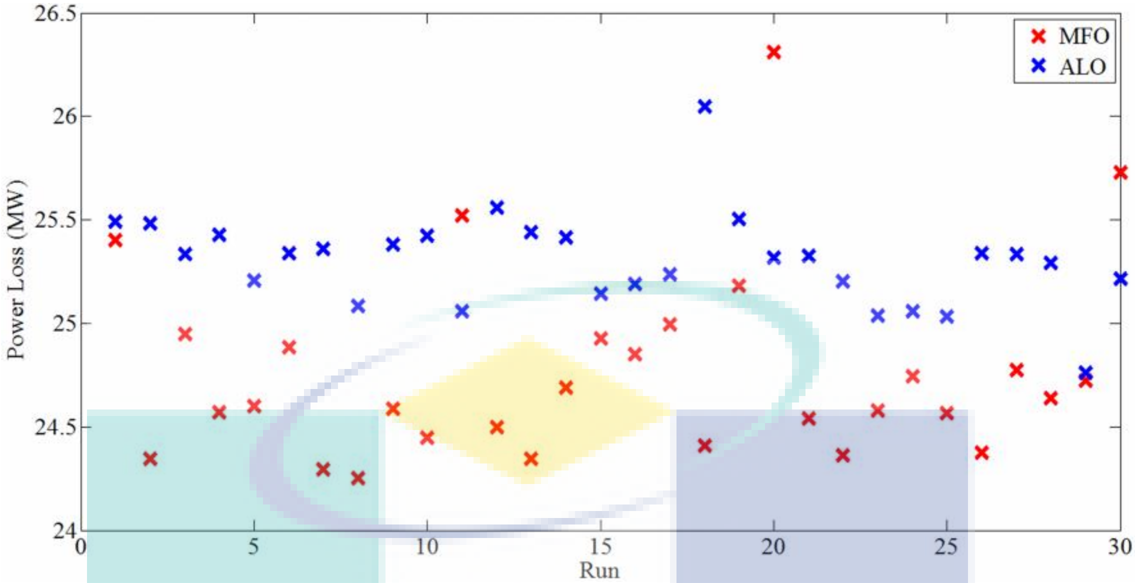

Figure 4.10 Comparison of power loss performances between ALO and MFO for 30 free running simulations (test case 4, IEEE 57-bus system)

4.6 Test Case 5: IEEE 118-Bus System with 77 Control Variables

In this test case, a standard IEEE 118-bus system is considered in order to test the ability of proposed MFO on solving bigger complex system. This test system constitutes a total of 77 control variables that need to be optimized. These involved 54 generator buses, nine tap setting transformers and 14 reactive compensator devices. Additionally, the IEEE 118-bus system data including line data, bus data and initial values of control variables are obtained from (Zimmerman et al., 2011). The minimum and maximum boundary setting of control variables is stated in Table 4.25. Since the maximum limits of Q_{C5} and Q_{C37} are zeros and their minimum limits are -40 and -25, respectively, the reactive power injection at bus 5 and 37 are set as 0 MVar. The system load demands of this system are given as follows: P_{Load} = 4242 MW and Q_{Load} = 1438 MVar. Moreover, it is vital to mention that the simulation parameters for both DE and GA are set according to the values stated in Table 4.4. In this test case, the NFE for MFO to reach the optimal solutions is 30000.

Table 4.25 Setting of variables limits for test case 5, IEEE 118-bus system

Control Variables	Lower Bound	Upper Bound
Generator Buses Voltage	0.95 p.u	1.10 p.u
Load Buses Voltage	0.95 p.u	1.05 p.u
Transformers Tap Setting	0.90 p.u	1.10 p.u
Q_{C5}	-40 MVar	0 MVar
Q_{C34}	0 MVar	14 MVar
Q_{C37}	-15 MVar	0 MVar
Q_{C44}	0 MVar	10 MVar
Q_{C45}	0 MVar	10 MVar
Q_{C46}	0 MVar	10 MVar
Q_{C48}	0 MVar	10 MVar
Q_{C74}	0 MVar	12 MVar
Q_{C79}	0 MVar	20 MVar
Q_{C82}	0 MVar	20 MVar
Q_{C83}	0 MVar	10 MVar
Q_{C105}	0 MVar	20 MVar
Q_{C107}	0 MVar	6 MVar
Q_{C110}	0 MVar	6 MVar

MFO based ORPD schedule for power loss minimization objective for this test case is reported in Table 4.26 together with the results optimized by DE and GA. From this table, it is worth to note that the results for the optimal setting of control variables obtained by proposed MFO are all converged within the specified ranges. Table 4.27 presents the comparison of the optimal power loss obtained by various algorithms. Based on this table, it can be noticed that MFO yields the lowest optimal power loss as compared to other reviewed algorithms. Table 4.28 tabulated that a power loss reduction of 12.37 % from the base case loss of 132.863 MW is accomplished by utilizing proposed MFO. It is the highest percentage of power loss reduction among other reported optimization approaches. In addition, the results also clearly proved the superiority of MFO over the best well-recognized algorithms (PSO, DE and GA). Moreover, Table 4.28 also reports the comparison of simulation time in seconds for different reviewed algorithms. Although MFO is less efficient in terms of computational time, it produces high quality results and its success rate is consistent as well.

Table 4.26 Optimal results of the control variables after optimization for test case 5, IEEE 118-bus system

Control Variables	**PSO**	**CLPSO**	**GSA**	**CPVEIHBMO**
Generator voltage				
V_1	1.0853	1.0332	0.9600	0.9926
V_4	1.0420	1.0550	0.9620	1.0108
V_6	1.0805	0.9754	0.9729	1.0037
V_8	0.9683	0.9669	1.0570	0.9976
V_{10}	1.0756	0.9811	1.0885	1.0215
V_{12}	1.0225	1.0092	0.9630	1.0093
V_{15}	1.0786	0.9787	1.0127	1.0075
V_{18}	1.0498	1.0799	1.0069	1.0259
V_{19}	1.0776	1.0805	1.0003	0.9943
V_{24}	1.0827	1.0286	1.0105	1.0179
V_{25}	0.9564	1.0307	1.0102	1.0177
V_{26}	1.0809	0.9877	1.0401	0.9990
V_{27}	1.0874	1.0157	0.9809	1.0084
V_{31}	0.9608	0.9615	0.9500	0.9838
V_{32}	1.1000	0.9851	0.9552	0.9827
V_{34}	0.9611	1.0157	0.9910	1.0065
V_{36}	1.0367	1.0849	1.0091	1.0190
V_{40}	1.0914	0.9830	0.9505	1.0267
V_{42}	0.9701	1.0516	0.9500	0.9865
V_{46}	1.0390	0.9754	0.9814	1.0084
V_{49}	1.0836	0.9838	1.0444	1.0035
V_{54}	0.9764	0.9637	1.0379	0.9806
V_{55}	1.0103	0.9716	0.9907	0.9969
V_{56}	0.9536	1.0250	1.0333	0.9881
V_{59}	0.9672	1.0003	1.0099	1.0197
V_{61}	1.0938	1.0771	1.0925	0.9956
V_{62}	1.0978	1.0480	1.0393	1.0064
V_{65}	1.0892	0.9684	0.9998	0.9883
V_{66}	1.0861	0.9648	1.0355	1.0101
V_{69}	0.9665	0.9574	1.1000	0.9931
V_{70}	1.0783	0.9765	1.0992	1.0127
V_{72}	0.9506	1.0243	1.0014	1.0145
V_{73}	0.9722	0.9651	1.0111	1.0174
V_{74}	0.9713	1.0733	1.0476	1.0025
V_{76}	0.9602	1.0302	1.0211	0.9842
V_{77}	1.0781	1.0275	1.0187	0.9914
V_{80}	1.0788	0.9857	1.0462	1.0257
V_{85}	0.9568	0.9836	1.0491	0.9876
V_{87}	0.9642	1.0882	1.0426	1.0213
V_{89}	0.9748	0.9895	1.0955	1.0069
V_{90}	1.0248	0.9905	1.0417	1.0298
V_{91}	0.9615	1.0288	1.0032	0.9839
V_{92}	0.9568	0.9760	1.0927	1.0021
V_{99}	0.9540	1.0880	1.0433	0.9853
V_{100}	0.9584	0.9617	1.0786	1.0281
V_{103}	1.0162	0.9611	1.0266	0.9802
V_{104}	1.0992	1.0125	0.9808	1.0187
V_{105}	0.9694	1.0684	1.0163	1.0209

Table 4.26 Continued

Control Variables	PSO	CLPSO	GSA	CPVEIHBMO
Generator voltage				
V_{107}	0.9656	0.9769	0.9987	1.0234
V_{110}	1.0873	1.0414	1.0218	0.9842
V_{111}	1.0375	0.9790	0.9852	1.0000
V_{112}	1.0920	0.9764	0.9500	0.9930
V_{113}	1.0753	0.9721	0.9764	1.0200
V_{116}	0.9594	1.0330	1.0372	1.0016
Transformer tap ratio				
T_{8-5}	1.0112	1.0045	1.0659	1.0255
T_{26-25}	1.0906	1.0609	0.9534	0.9891
T_{30-17}	1.0033	1.0008	0.9328	0.9932
T_{38-37}	1.0000	1.0093	1.0884	0.9873
T_{63-59}	1.0080	0.9922	1.0579	0.9868
T_{64-61}	1.0326	1.0074	0.9493	1.0235
T_{65-66}	0.9443	1.0611	0.9975	1.0090
T_{68-69}	0.9067	0.9307	0.9887	1.0075
T_{81-80}	0.9673	0.9578	0.9801	0.9872
Capacitor bank				
Q_{C5}	0	0	0	0
Q_{C34}	9.3639	11.7135	7.4600	6.0111
Q_{C37}	0	0	0	0
Q_{C44}	9.3078	9.8932	6.0700	6.0057
Q_{C45}	8.6428	9.4169	3.3300	3.0001
Q_{C46}	8.9462	2.6719	6.5100	5.9838
Q_{C48}	11.8092	2.8546	4.4700	3.9920
Q_{C74}	4.6132	0.5471	9.7200	7.9862
Q_{C79}	10.5923	14.8532	14.2500	13.9892
Q_{C82}	16.4544	19.4270	17.4900	17.9920
Q_{C83}	9.6325	6.9824	4.2800	4.0009
Q_{C105}	8.9513	9.0291	12.0400	10.9825
Q_{C107}	5.0426	4.9926	2.2600	2.0251
Q_{C110}	5.5319	2.2086	2.9400	2.0272
P_{Loss} (MW)	131.9900	130.9600	127.7603	124.0983

Table 4.26 Continued

Control Variables	GWO	DE	GA	MFO
Generator voltage				
V_1	1.0204	1.0336	1.0130	1.0173
V_4	1.0257	1.0474	1.0221	1.0402
V_6	1.0208	1.0316	1.0200	1.0292
V_8	1.0419	1.0334	1.0615	1.0600
V_{10}	1.0413	1.0347	1.0994	1.0374
V_{12}	1.0232	1.0433	1.0148	1.0250
V_{15}	1.0207	1.0266	1.0196	1.0268
V_{18}	1.0270	1.0272	1.0432	1.0298
V_{19}	1.0204	1.0307	1.0295	1.0275
V_{24}	1.0137	1.0319	1.0405	1.0483
V_{25}	1.0270	1.0435	1.0759	1.0600
V_{26}	1.0386	1.0104	1.0572	1.0600
V_{27}	1.0188	1.0189	1.0292	1.0267
V_{31}	1.0138	1.0481	1.0215	1.0101
V_{32}	1.0135	1.0215	1.0271	1.0226
V_{34}	1.0261	1.0277	1.0414	1.0556
V_{36}	1.0261	1.0254	1.0361	1.0548
V_{40}	1.0125	1.0224	1.0350	1.0419
V_{42}	1.0233	1.0226	1.0274	1.0429
V_{46}	1.0272	1.0245	1.0083	1.0450
V_{49}	1.0401	1.0426	1.0408	1.0589
V_{54}	1.0230	1.0135	1.0298	1.0284
V_{55}	1.0221	1.0153	1.0289	1.0289
V_{56}	1.0226	1.0131	1.0247	1.0283
V_{59}	1.0379	1.0405	1.0473	1.0512
V_{61}	1.0241	1.0249	1.0583	1.0534
V_{62}	1.0199	1.0161	1.0521	1.0506
V_{65}	1.0465	1.0414	1.0477	1.0596
V_{66}	1.0378	1.0563	1.0531	1.0600
V_{69}	1.0501	1.0571	1.0439	1.0600
V_{70}	1.0243	1.0323	1.0245	1.0600
V_{72}	1.0187	1.0454	1.0252	1.0526
V_{73}	1.0397	1.0331	1.0415	1.0600
V_{74}	1.0170	1.0374	1.0196	1.0600
V_{76}	1.0080	1.0407	1.0134	1.0390
V_{77}	1.0192	1.0438	1.0126	1.0502
V_{80}	1.0329	1.0468	1.0210	1.0600
V_{85}	1.0224	1.0206	1.0273	1.0600
V_{87}	1.0361	1.0206	1.0155	1.0599
V_{89}	1.0558	1.0436	1.0612	1.0600
V_{90}	1.0290	1.0166	1.0490	1.0431
V_{91}	1.0127	1.0146	1.0500	1.0496
V_{92}	1.0360	1.0374	1.0370	1.0600
V_{99}	1.0297	1.0034	1.0332	1.0551
V_{100}	1.0360	1.0384	1.0251	1.0584
V_{103}	1.0232	1.0450	1.0084	1.0442
V_{104}	1.0180	1.0459	1.0219	1.0333
V_{105}	1.0176	1.0383	1.0102	1.0281
V_{107}	1.0201	1.0141	0.9929	1.0161

Table 4.26 Continued

Control Variables	GWO	DE	GA	MFO
Generator voltage				
V_{110}	1.0207	1.0518	0.9982	1.0215
V_{111}	1.0261	1.0342	1.0103	1.0280
V_{112}	1.0066	1.0454	0.9908	1.0042
V_{113}	1.0251	1.0281	1.0356	1.0350
V_{116}	1.0342	1.0508	1.0239	1.0484
Transformer tap ratio				
T_{8-5}	1.0208	0.9937	1.0206	1.0136
T_{26-25}	1.0279	1.0081	0.9910	1.1000
T_{30-17}	1.0323	0.9789	0.9939	1.0038
T_{38-37}	1.0209	1.0169	1.0014	0.9826
T_{63-59}	1.0091	0.9973	0.9882	0.9843
T_{64-61}	1.0366	1.0258	0.9598	1.0139
T_{65-66}	1.0301	1.0342	0.9804	1.1000
T_{68-69}	1.0234	0.9873	0.9011	1.1000
T_{81-80}	1.0211	0.9930	0.9829	0.9683
Capacitor bank				
Q_{C5}	-39.7600	-16.3153	-19.7569	0
Q_{C34}	13.7900	7.9425	8.7062	0
Q_{C37}	-24.7300	-9.4528	-8.7876	-0.0313
Q_{C44}	9.9571	5.8755	9.8657	10.0000
Q_{C45}	9.8678	5.0360	8.1047	0
Q_{C46}	9.9186	3.5833	4.3640	0
Q_{C48}	14.8900	4.7675	5.1546	0.0008
Q_{C74}	11.9720	6.9687	2.3461	0.2205
Q_{C79}	19.6490	10.2978	11.8827	20.0000
Q_{C82}	19.8900	11.6685	10.2995	0
Q_{C83}	9.9515	4.0756	4.4286	10.0000
Q_{C105}	19.9680	5.0313	11.5653	0
Q_{C107}	5.9136	3.0884	3.1598	6.0000
Q_{C110}	5.8834	2.6946	2.8654	6.0000
P_{Loss} (MW)	120.6538	122.3603	119.3056	116.4254

Table 4.27 Comparison of optimal transmission loss for MFO and different optimization algorithms (test case 5, IEEE 118-bus system)

Optimization Algorithms	Minimum Power Loss (MW)
Particle Swarm Optimization, PSO (Mahadevan & Kannan, 2010)	131.9900
Comprehensive Learning Particle Swarm Optimization, CLPSO (Mahadevan & Kannan, 2010)	130.9600
Gravitational Search Algorithm, GSA (S Duman et al., 2012)	127.7603
Chaotic Parallel Vector Evaluated Interactive Honey Bee Mating Optimization, CPVEIHBMO (A. Ghasemi et al., 2014)	124.0983
Opposition-based Gravitational Search Algorithm, OGSA (Shaw et al., 2014)	126.9900
Exchange Market Algorithm, EMA (Rajan & Malakar, 2016)	126.2243
Particle Swarm Optimization with an Aging Leader and Challengers, ALC-PSO (Singh, Mukherjee, & Ghoshal, 2015)	121.5300
Hybrid Particle Swarm Optimization and Gravitational Search Algorithm, PSO-GSA (Radosavljević et al., n.d.)	122.4709
Grey Wolf Optimizer, GWO (Mohd Herwan Sulaiman et al., 2015)	120.6538
Differential Evolution, DE	122.3603
Genetic Algorithm, GA	119.3056
Moth-Flame Optimization, MFO	116.4254

Table 4.28 Comparison of percentage of loss reduction and computational time by MFO and other optimization algorithms (test case 5, IEEE 118-bus system)

Compared Items	Power Loss, P_{Loss} (MW)	Percentage of Loss Reduction (%)	Simulation Time (s)
Base Case	132.8630	N/A	N/A
PSO	131.9900	0.66	1215.0000
CLPSO	130.9600	1.43	1472.0000
GSA	127.7603	3.84	1198.6583
CPVEIHBMO	124.0983	6.60	1053.3725
OGSA	126.9900	4.42	1101.2600
EMA	126.2243	5.00	N/A
ALC-PSO	121.5300	8.53	N/A
PSO-GSA	122.4709	7.82	1045.1000
GWO	120.6538	9.19	1372.0000
DE	122.3603	7.90	N/A
GA	119.3056	10.20	N/A
MFO	116.4254	12.37	1419.0000

Table 4.29 demonstrates the statistical comparison of power loss minimization results between MFO and other reviewed techniques. Obviously, proposed MFO is able to produce the best and average solutions among others. Nevertheless, it also produces the worst result. It is worth to emphasize that all the solutions of MFO in terms of best, average and worst results are obtained from 30 trail runs. On the other hand, the results of the other reviewed algorithms are obtained through 100 trail runs. Figure 4.11 shows the performance characteristics of proposed MFO using 30 search agents for 30 simulations. Figure 4.12 illustrates the comparative convergence performance of power loss yielded by three different numbers of search agents against iterations. Good convergence profile yielded by 30 search agents can be noted from this graph by means of their fastest convergence and ability to reach the optimal solution.

Table 4.29 Statistical comparison of results based with power loss minimization objective (test case 5, IEEE 118-bus system)

Compared Item (P_{Loss})	PSO	CLPSO	ALC-PSO	OGSA	MFO
Best Result (MW)	131.9900	130.9600	121.5300	126.9900	116.4400
Average Result (MW)	132.3700	131.1500	123.1400	127.1400	123.9207
Worst Result (MW)	134.5000	132.7400	132.9900	131.9900	135.9600

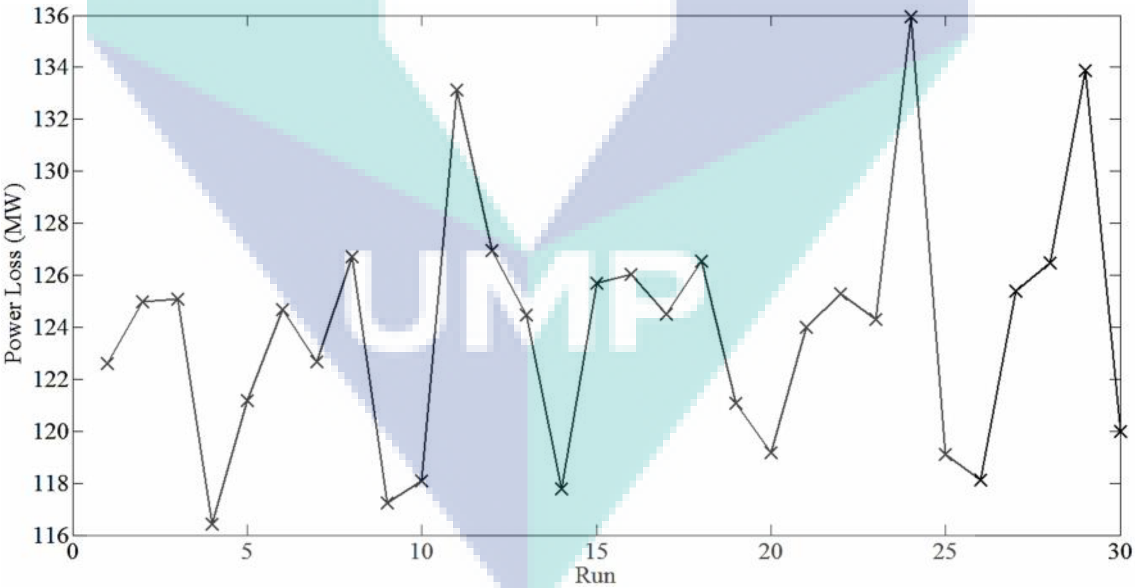

Figure 4.11 Performance characteristics of MFO using 30 search agents for 30 free running simulations (test case 5, IEEE 118-bus system)

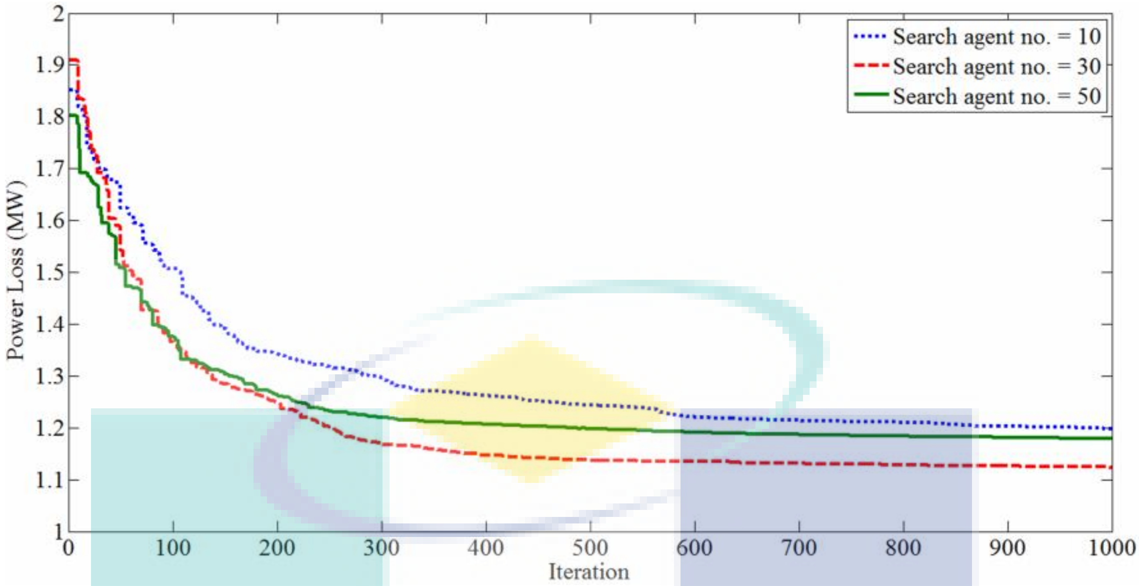

Figure 4.12 Convergence performances of power loss of IEEE 118-bus system for different numbers of search agents using MFO algorithm

For voltage deviation minimization objective, the best ORPD solutions yielded by proposed MFO from 30 simulations are tabulated in Table 4.30. The minimized voltage deviation results of MFO are compared with those results obtained by OGSA and ALC-PSO. Obviously, it may be noticed that the value of voltage deviation for proposed MFO-based approach is the most promising one among others. The convergence characteristics of MFO for voltage deviation minimization over the course of iterations are illustrated in Figure 4.13. Figure 4.14 demonstrates the performance characteristics of MFO for the second objective function using 30 search agents over 30 free running simulations.

Table 4.30 Comparison of simulation results of voltage deviation minimization objective for test case 5, IEEE 118-bus system

Control Variables	OGSA (Shaw et al., 2014)	ALC-PSO (Singh et al., 2015)	MFO
Generator voltage			
V_1	1.0388	1.0011	1.0025
V_4	0.9872	1.0191	0.9858
V_6	0.9925	0.9934	0.9916
V_8	0.9905	0.9762	1.0312
V_{10}	0.9919	1.0064	0.9500
V_{12}	1.0077	1.0126	1.0155
V_{15}	1.0034	0.9865	1.0083
V_{18}	0.9773	1.0560	1.1000
V_{19}	1.0324	1.0188	1.0197
V_{24}	1.0285	1.0202	1.0009
V_{25}	0.9705	1.0111	1.0056
V_{26}	1.0175	0.9801	0.9500
V_{27}	1.0117	1.0231	1.0136
V_{31}	1.0014	0.9994	1.0018
V_{32}	0.9988	0.9878	0.9983
V_{34}	1.0158	1.0214	1.0082
V_{36}	0.9916	0.9655	0.9999
V_{40}	1.0132	1.0043	1.0063
V_{42}	0.9892	1.0138	1.0084
V_{46}	1.0607	1.0526	1.0407
V_{49}	1.0031	1.0024	1.0026
V_{54}	1.0236	1.0234	1.0232
V_{55}	1.0176	1.0330	1.1000
V_{56}	1.0149	1.0146	1.0155
V_{59}	1.0584	1.0090	1.1000
V_{61}	0.9829	1.0003	0.9992
V_{62}	1.0562	1.0037	0.9750
V_{65}	0.9724	0.9688	0.9529
V_{66}	1.0020	1.0165	1.0337
V_{69}	0.9827	1.0438	0.9500
V_{70}	0.9997	0.9735	1.0187
V_{72}	1.0123	0.9970	1.1000
V_{73}	0.9960	1.0335	0.9500
V_{74}	1.0232	1.0028	1.0231
V_{76}	1.0015	1.0030	1.0130
V_{77}	1.0124	1.0255	1.0051
V_{80}	1.0226	0.9924	1.0179
V_{85}	1.0117	1.0206	1.0097
V_{87}	1.0058	0.9900	1.0084
V_{89}	1.0076	1.0017	1.0085
V_{90}	0.9753	1.0792	1.1000
V_{91}	0.9836	0.9930	0.9500
V_{92}	1.0272	0.9989	1.0146
V_{99}	0.9612	1.0682	1.1000
V_{100}	1.0032	1.0424	1.0140
V_{103}	0.9843	1.0509	0.9500
V_{104}	0.9880	0.9864	1.0980

Table 4.30 Continued

Control Variables	OGSA (Shaw et al., 2014)	ALC-PSO (Singh et al., 2015)	MFO
Generator voltage			
V_{105}	1.0003	0.9993	1.0045
V_{107}	1.0033	1.0219	1.0220
V_{110}	1.0040	1.0048	1.0008
V_{111}	1.0331	1.0581	1.0996
V_{112}	0.9877	1.0178	1.1000
V_{113}	0.9705	0.9826	0.9500
V_{116}	1.0270	0.9929	1.0337
Transformer tap ratio			
$T_{8\text{-}5}$	0.9841	1.0491	0.9846
$T_{26\text{-}25}$	1.0377	0.9499	1.1000
$T_{30\text{-}17}$	0.9573	1.0328	0.9712
$T_{38\text{-}37}$	0.9952	0.9814	1.0024
$T_{63\text{-}59}$	0.9622	1.0223	0.9000
$T_{64\text{-}61}$	1.0320	0.9972	1.0512
$T_{65\text{-}66}$	1.0137	1.0257	1.1000
$T_{68\text{-}69}$	0.9795	0.9667	0.9000
$T_{81\text{-}80}$	0.9985	1.0124	0.9000
Capacitor bank			
Q_{C5}	-0.2403	-0.1438	-40.0000
Q_{C34}	0.0371	0.0205	14.0000
Q_{C37}	-0.0437	-0.1385	0
Q_{C44}	0.0375	0.0778	10.0000
Q_{C45}	0.0400	0.0454	10.0000
Q_{C46}	0.0749	0.0544	10.0000
Q_{C48}	0.0796	0.1027	0
Q_{C74}	0.0883	0.0085	12.0000
Q_{C79}	0.1218	0.0187	20.0000
Q_{C82}	0.0380	0.1428	20.0000
Q_{C83}	0.0627	0.0837	10.0000
Q_{C105}	0.0830	0.1175	20.0000
Q_{C107}	0.0459	0.0260	0
Q_{C110}	0.0221	0.0246	0
Voltage deviations (p.u)	0.3666	0.3262	0.2147
Simulation time (s)	1121.1700	1111.2600	1186.2010

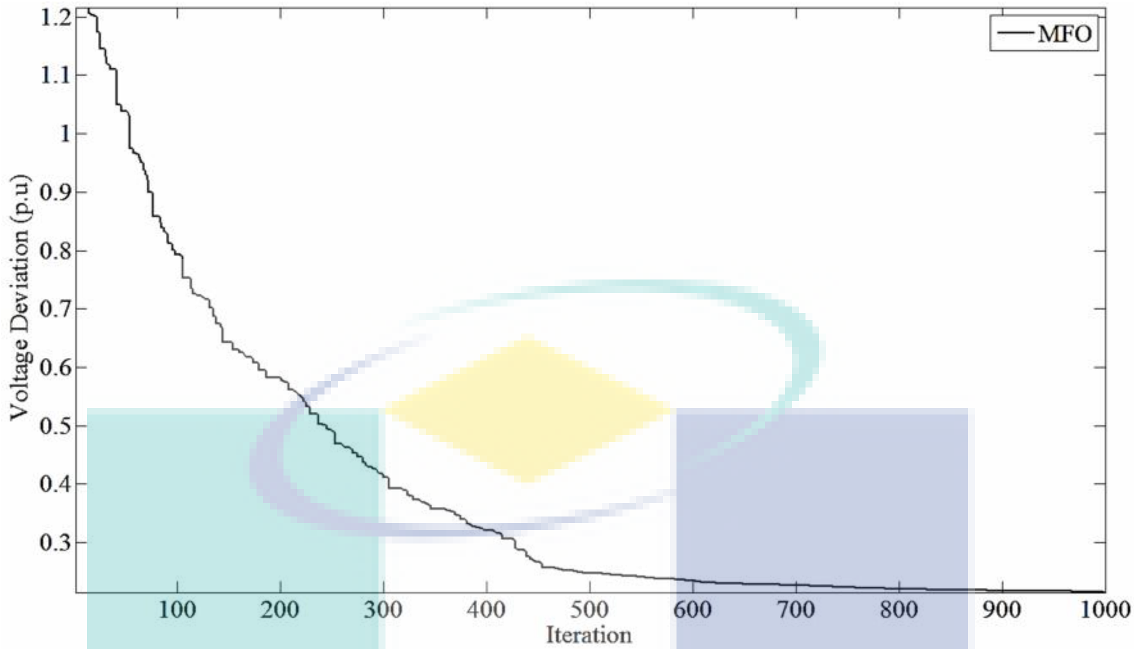

Figure 4.13 Convergence characteristic of MFO for voltage deviation minimization (test case 5, IEEE 118-bus system)

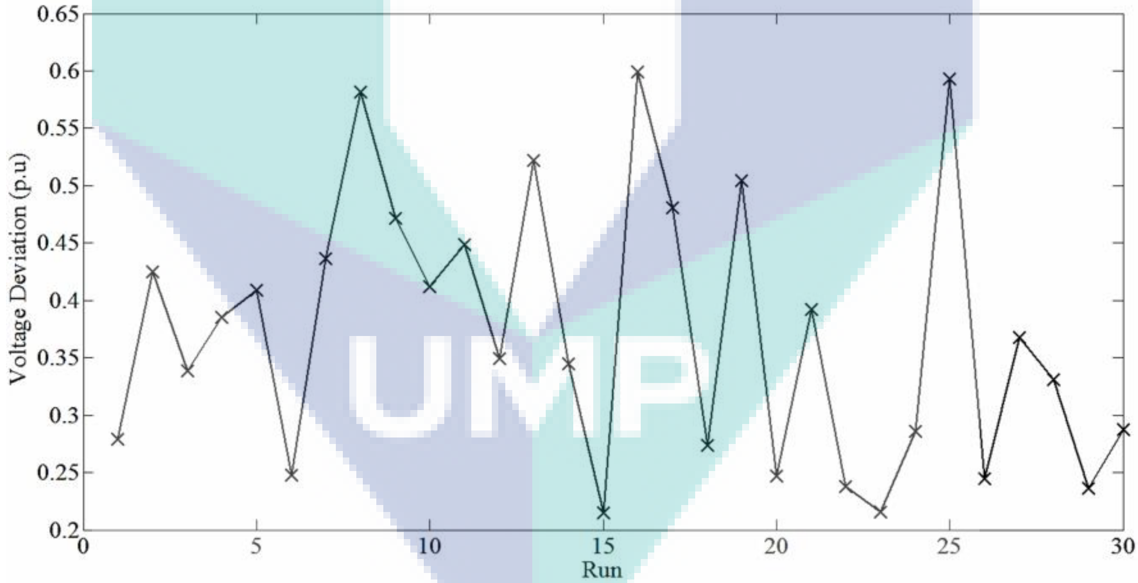

Figure 4.14 Performance characteristics of MFO for voltage deviation minimization using 30 search agents for 30 free running simulations (test case 5, IEEE 118-bus system)

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

This chapter summarizes the achievements of this research study, including its contributions and recommendations for future work. All the objectives of this research aforementioned in chapter 1 have been achieved through the simulation results.

In this research, the implementation of a new nature-inspired meta-heuristic approach namely moth-flame optimization (MFO) algorithm has been used to solve nonlinear ORPD problems. Five case studies are conducted to validate the effectiveness and convergence performance of this algorithm, including IEEE 30-bus system (with 13, 19 and 25 control variables), IEEE 57-bus system and IEEE 118-bus system. It is worth to highlight that another new meta-heuristic technique namely ant lion optimizer (ALO) also included in this study. The performance of ALO was evaluated using three test cases, including IEEE 30-bus system (with 13 and 25 control variables) and IEEE 57-bus system. Its simulation results are compared against proposed MFO. These two algorithms demonstrate their ability in obtaining near optimal solutions on solving ORPD problems. When compared the simulation results between these two techniques, proposed MFO seems to be more effective as the total transmission loss and voltage deviation are the minimum relative to ALO.

Undeniably, proposed MFO also proved its superiority against the best results of other techniques reported in the recent state-of-the-art literatures. These included the best well-known methods (PSO, DE and GA), recently developed algorithms (GWO, MVO and EMA) as well as hybrid methods (MICA-IWO, ALC-PSO and PSO-GSA). It offered novel and outstanding solutions with the lowest system power loss and voltage deviations for the five case studies. It demonstrates its stable performance in addressing nonlinear characteristic of ORPD problems with continuous and discrete control variables. Despite of the size of the test system, it performs well on both basic test system and larger test system with more control variables. Additionally, proposed MFO possess high robustness and it can prevent local optima entrapment. In this research, it is vital to emphasize that no specific simulation parameter is needed to be pre-set for both MFO and ALO except number of search agents and maximum iterations. This makes them superior in terms of simplicity.

Nevertheless, the proposed MFO has its weakness which it requires longer computational time to reach the optimal solution. This can be seen from the simulation results of test case 5 where it possesses the highest simulation time among others. Since the computational time calculated in this research is in seconds (s), the weakness of proposed MFO can be neglected. In this research, a promising MATPOWER 5.1 toolbox is implemented in order to analyse the optimized results fairly. Moreover, penalty function method also being applied in order to avoid the optimized control variables exceeding the limit constraints.

5.2 Contribution

The main contributions of this research towards ORPD are summarized as below:

1. A newly developed nature-inspired optimization algorithm namely MFO has been implemented to solve nonlinear ORPD problems. As expected, this algorithm validated its superior ability on solving the problem by producing the lowest power loss and voltage deviation among other rival approaches. These achievements will further benefit the economic dispatch and secure operation of the power system. Moreover, the proposed MFO has not yet been applied in solving ORPD problems as reviewed in the latest literatures. Since MFO is outperform on solving nonlinear optimization problem, its strength can be identified which might be useful to solve other optimization problems in any other areas.

2. In terms of research objectives achievement, proposed MFO successfully improves ORPD problems by its spiral convergence characteristic. Additionally, MFO is also able to minimize the total transmission loss and voltage deviation

by finding the optimal setting of control variables. For all the test cases presented, proposed MFO demonstrated outperform results by producing optimum minimum solutions for both power loss and voltage deviation. This achievement in reduction of loss and voltage deviation helps to handle the critical problems of power system by dispatching the load based on its demand while without influence the reliability of the generators.

3. Implementation of mature MATPOWER 5.1 software package to execute the power flow and fulfil the equality constraints in MFO based ORPD problems. By utilizing this standard and stable toolbox, the accuracy of the solutions can be guaranteed and improved. Besides, penalty function method also applied in this research to avoid the control variables exceeding the violation of the constraints.

5.3 Recommendation for Future Research

Application of MFO based ORPD problems opens new framework for further research, such as the following recommendations:

1. Improving the original version of MFO algorithm or proposing hybrid method to enhance the robustness, efficiency and quality of MFO algorithm.

2. Further employment on solving ORPD with practical operating constraints related to generating units, including prohibited zones and valve points loading effects.

3. Consideration of MFO algorithm on solving multi-objective ORPD problem that involving both transmission loss and voltage deviation simultaneously.

REFERENCES

Abido, M. A. (2002). Optimal power flow using particle swarm optimization. *International Journal of Electrical Power & Energy Systems*, *24*(7), 563–571. JOUR.

Akorede, M. F., & Hizam, H. (2009). Teaching power system analysis courses using MATPOWER. In *Engineering Education (ICEED), 2009 International Conference on* (pp. 45–51). CONF, IEEE.

Allam, D., Yousri, D. A., & Eteiba, M. B. (2016). Parameters extraction of the three diode model for the multi-crystalline solar cell/module using Moth-Flame Optimization Algorithm. *Energy Conversion and Management*, *123*, 535–548. JOUR.

Alsac, O., Bright, J., Prais, M., & Stott, B. (1990). Further developments in LP-based optimal power flow. *IEEE Transactions on Power Systems*, *5*(3), 697–711. JOUR.

Ayan, K., & Kılıç, U. (2012). Artificial bee colony algorithm solution for optimal reactive power flow. *Applied Soft Computing*, *12*(5), 1477–1482. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.asoc.2012.01.006

Bakirtzis, A. G., Biskas, P. N., Zoumas, C. E., & Petridis, V. (2002). Optimal power flow by enhanced genetic algorithm. *IEEE Transactions on Power Systems*, *17*(2), 229–236. JOUR.

Bentouati, B., Chaib, L., & Chettih, S. (2016). Optimal Power Flow using the Moth Flam Optimizer: A case study of the Algerian power system. *Indonesian Journal of Electrical Engineering and Computer Science*, *1*(3), 431–445. JOUR.

Bianchi, L., Dorigo, M., Gambardella, L. M., & Gutjahr, W. J. (2009). A survey on metaheuristics for stochastic combinatorial optimization. *Natural Computing: An International Journal*, *8*(2), 239–287. JOUR.

Blasco, X. (1999). Control predictivo basado en modelos mediante técnicas de optimización heurística. Aplicación a procesos no lineale sy multivariables. DISS, Tesis Doctoral. Universidad Politécnica de Valencia, España.

Blum, C., Puchinger, J., Raidl, G. R., & Roli, A. (2011). Hybrid metaheuristics in combinatorial optimization: A survey. *Applied Soft Computing*, *11*(6), 4135–4151. JOUR.

BoussaïD, I., Lepagnot, J., & Siarry, P. (2013). A survey on optimization metaheuristics. *Information Sciences*, *237*, 82–117. JOUR.

Ceylan, O., Ozdemir, A., & Dag, H. (2010). Gravitational search algorithm for post-outage bus voltage magnitude calculations. In *45th International Universities Power Engineering*

Conference UPEC2010. CONF.

Chebbo, A. M., Irving, M. R., & Sterling, M. J. H. (1992). Reactive power dispatch incorporating voltage stability. *IEE Proceedings C - Generation, Transmission and Distribution*, *139*(3), 253–260. Journal Article. https://doi.org/10.1049/ip-c.1992.0038

Chen, G., Liu, L., Zhang, Z., & Huang, S. (2017). Optimal reactive power dispatch by improved GSA-based algorithm with the novel strategies to handle constraints. *Applied Soft Computing*, *50*, 58–70. JOUR.

Dai, C., Chen, W., Zhu, Y., & Zhang, X. (2009). Seeker optimization algorithm for optimal reactive power dispatch. *IEEE Transactions on Power Systems*, *24*(3), 1218–1231. JOUR.

Das, S., Abraham, A., & Konar, A. (2008). Particle swarm optimization and differential evolution algorithms: technical analysis, applications and hybridization perspectives. In *Advances of Computational Intelligence in Industrial Systems* (pp. 1–38). CHAP, Springer.

Devaraj, D. (2007). Improved genetic algorithm for multi- objective reactive power dispatch problem. *European Transactions on Electrical Power*, *17*(6), 569–581. JOUR.

Devaraj, D., Durairaj, S., & Kannan, P. S. (2008). Real parameter genetic algorithm for multiobjective reactive power dispatch. *International Journal of Power & Energy Systems*, *28*(1), 41. JOUR.

Devaraj, D., & Roselyn, J. P. (2010). Genetic algorithm based reactive power dispatch for voltage stability improvement. *International Journal of Electrical Power & Energy Systems*, *32*(10), 1151–1156. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.ijepes.2010.06.014

Dommel, H. W., & Tinney, W. F. (1968). Optimal power flow solutions. *IEEE Transactions on Power Apparatus and Systems*, (10), 1866–1876. JOUR.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *26*(1), 29–41. JOUR.

Duman, S., Güvenç, U., Sönmez, Y., & Yörükeren, N. (2012). Optimal power flow using gravitational search algorithm. *Energy Conversion and Management*, *59*, 86–95. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.enconman.2012.02.024

Duman, S., Sonmez, Y., Guvenc, U., & Yorukeren, N. (2012). Optimal reactive power dispatch using a gravitational search algorithm. *IET Generation, Transmission & Distribution*, *6*(6), 563–576. JOUR.

Durairaj, S., Devaraj, D., & Kannan, P. S. (2006). Genetic algorithm applications to optimal reactive power dispatch with voltage stability enhancement. *Journal-Institution of Engineers India Part El Electrical Engineering Division*, *87*, 42. JOUR.

Dutta, S., Paul, S., & Roy, P. K. (2016). Optimal allocation of SVC and TCSC using Quasi-oppositional chemical reaction optimization for solving multi-objective ORPD problem. *Journal of Electrical Systems and Information Technology*. JOUR.

Eberhart, R. C., & Kennedy, J. (1995). A new optimizer using particle swarm theory. In *Proceedings of the sixth international symposium on micro machine and human science* (Vol. 1, pp. 39–43). CONF, New York, NY.

El Ela, A. A. A., Abido, M. A., & Spea, S. R. (2011). Differential evolution algorithm for optimal reactive power dispatch. *Electric Power Systems Research*, *81*(2), 458–464. JOUR.

Frank, K. D., Rich, C., & Longcore, T. (2006). Effects of artificial night lighting on moths. *Ecological Consequences of Artificial Night Lighting*, 305–344. JOUR.

Geem, Z. W., Kim, J. H., & Loganathan, G. V. (2001). A new heuristic optimization algorithm: harmony search. *Simulation*, *76*(2), 60–68. JOUR.

Ghasemi, A., Valipour, K., & Tohidi, A. (2014). Multi objective optimal reactive power dispatch using a new multi objective strategy. *International Journal of Electrical Power & Energy Systems*, *57*, 318–334. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.ijepes.2013.11.049

Ghasemi, M., Ghavidel, S., Ghanbarian, M. M., & Habibi, A. (2014). A new hybrid algorithm for optimal reactive power dispatch problem with discrete and continuous control variables. *Applied Soft Computing*, *22*, 126–140. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.asoc.2014.05.006

Gogna, A., & Tayal, A. (2013). Metaheuristics: review and application. *Journal of Experimental & Theoretical Artificial Intelligence*, *25*(4), 503–526. JOUR.

Golberg, D. E. (1989). Genetic algorithms in search, optimization, and machine learning. *Addion Wesley*, *1989*, 102. JOUR.

Gomes, J. R., & Saavedra, O. R. (2002). A Cauchy-based evolution strategy for solving the reactive power dispatch problem. *International Journal of Electrical Power & Energy Systems*, *24*(4), 277–283. JOUR.

Granville, S. (1994). Optimal reactive dispatch through interior point methods. *IEEE Transactions on Power Systems*, *9*(1), 136–146. JOUR.

Grudinin, N. (1998). Reactive power optimization using successive quadratic programming method. *IEEE Transactions on Power Systems*, *13*(4), 1219–1225. JOUR.

Haddad, O. B., Afshar, A., & Mariño, M. A. (2006). Honey-bees mating optimization (HBMO) algorithm: a new heuristic approach for water resources optimization. *Water Resources Management*, *20*(5), 661–680. JOUR.

Happ, H. H. (1977). Optimal power dispatch꜍A comprehensive survey. *IEEE Transactions on Power Apparatus and Systems*, *96*(3), 841–854. JOUR.

Hoos, H. H., & Stützle, T. (2004). *Stochastic local search: Foundations & applications*. BOOK, Elsevier.

Huang, C.-M., & Huang, Y.-C. (2012). Combined differential evolution algorithm and ant system for optimal reactive power dispatch. *Energy Procedia*, *14*, 1238–1243. JOUR.

Iba, K. (1994). Reactive power optimization by genetic algorithm. *IEEE Transactions on Power Systems*, *9*(2), 685–692. JOUR.

Jangir, N., Pandya, M. H., Bhesdadiya, R. H., & Jangir, P. (n.d.). Price Penalty factors Based Approach for Combined Economic Emission Dispatch Problem Solution using Moth-flame Optimizer Algorithm. JOUR.

Jangir, N., Pandya, M. H., Trivedi, I. N., Bhesdadiya, R. H., Jangir, P., & Kumar, A. (2016). Moth-Flame Optimization algorithm for solving real challenging constrained engineering optimization problems. In *Electrical, Electronics and Computer Science (SCEECS), 2016 IEEE Students' Conference on* (pp. 1–5). CONF, IEEE.

Jangir, P., Parmar, S. A., Trivedi, I. N., & Bhesdadiya, R. H. (2016). A novel hybrid Particle Swarm Optimizer with multi verse optimizer for global numerical optimization and Optimal Reactive Power Dispatch problem. *Engineering Science and Technology, an International Journal*. JOUR.

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization* (RPRT). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of Global Optimization*, *39*(3), 459–471. JOUR.

KARABOĞA, D., & Ökdem, S. (2004). A simple and global optimization algorithm for engineering problems: differential evolution algorithm. *Turkish Journal of Electrical Engineering & Computer Sciences*, *12*(1), 53–60. JOUR.

Kaveh, A., & Farhoudi, N. (2013). A new optimization method: Dolphin echolocation. *Advances in Engineering Software*, *59*, 53–70. JOUR.

Khazali, A. H., & Kalantar, M. (2011). Optimal reactive power dispatch based on harmony search algorithm. *International Journal of Electrical Power & Energy Systems*, *33*(3), 684–692. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.ijepes.2010.11.018

Kirkpatrick, S. (1984). Optimization by simulated annealing: Quantitative studies. *Journal of Statistical Physics*, *34*(5–6), 975–986. JOUR.

Kirschen, D. S., & Van Meeteren, H. P. (1988). MW/voltage control in a linear programming based optimal power flow. *Power Systems, IEEE Transactions on*, *3*(2), 481–489. https://doi.org/10.1109/59.192899

Lai, L. L., & Ma, J. T. (1997). Application of evolutionary programming to reactive power planning-comparison with nonlinear programming approach. *IEEE Transactions on Power Systems*, *12*(1), 198–206. JOUR.

Lampinen, J., & Zelinka, I. (2000). On stagnation of the differential evolution algorithm. In *Proceedings of MENDEL* (pp. 76–83). CONF.

Lee, K. Y., Bai, X., & Park, Y.-M. (1995). Optimization method for reactive power planning by using a modified simple genetic algorithm. *IEEE Transactions on Power Systems*, *10*(4), 1843–1850. JOUR.

Lee, K. Y., & El-Sharkawi, M. A. (2008). *Modern heuristic optimization techniques: theory and applications to power systems* (Vol. 39). BOOK, John Wiley & Sons.

Lenin, K. (n.d.). UNIQUE ALGORITHM FOR SOLVING OPTIMAL REACTIVE POWER DISPATCH PROBLEM. JOUR.

Lenin, K., Reddy, B. R., & Kalavathi, M. S. (n.d.). IMPROVED BABOON ALGORITHM FOR MINIMIZATION OF REAL POWER LOSS. JOUR.

Li, Y., Cao, Y., Liu, Z., Liu, Y., & Jiang, Q. (2009). Dynamic optimal reactive power dispatch based on parallel particle swarm optimization algorithm. *Computers & Mathematics with Applications*, *57*(11), 1835–1842. JOUR.

Liang, C. H., Chung, C. Y., Wong, K. P., Duan, X. Z., & Tse, C. T. (2007). Study of differential evolution for optimal reactive power flow. *IET Generation, Transmission & Distribution*, *1*(2), 253–260. JOUR.

Lin, X., David, A. K., & Yu, C. W. (2003). Reactive power optimisation with voltage stability consideration in power market systems. *IEE Proceedings-Generation, Transmission and*

Distribution, *150*(3), 305–310. JOUR.

Ma, J. T., & Lai, L. L. (1996). Evolutionary programming approach to reactive power planning. *IEE Proceedings-Generation, Transmission and Distribution*, *143*(4), 365–370. JOUR.

Mahadevan, K., & Kannan, P. S. (2010). Comprehensive learning particle swarm optimization for reactive power dispatch. *Applied Soft Computing*, *10*(2), 641–652. JOUR.

Mallipeddi, R., Jeyadevi, S., Suganthan, P. N., & Baskar, S. (2012). Efficient constraint handling for optimal reactive power dispatch problems. *Swarm and Evolutionary Computation*, *5*, 28–36. JOUR.

Mehdinejad, M., Mohammadi-Ivatloo, B., Dadashzadeh-Bonab, R., & Zare, K. (2016). Solution of optimal reactive power dispatch of power systems using hybrid particle swarm optimization and imperialist competitive algorithms. *International Journal of Electrical Power & Energy Systems*, *83*, 104–116. JOUR.

Michalewicz, Z. (1996). Introduction. In *Genetic Algorithms+ Data Structures= Evolution Programs* (pp. 1–10). CHAP, Springer.

Miranda, V., Srinivasan, D., & Proenca, L. M. (1998). Evolutionary computation in power systems. *International Journal of Electrical Power & Energy Systems*, *20*(2), 89–98. JOUR.

Mirjalili, S. (2015a). Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowledge-Based Systems*, *89*, 228–249. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.knosys.2015.07.006

Mirjalili, S. (2015b). The Ant Lion Optimizer. *Advances in Engineering Software*, *83*, 80–98. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.advengsoft.2015.01.010

Mirjalili, S., Mirjalili, S. M., & Lewis, A. (2014). Grey wolf optimizer. *Advances in Engineering Software*, *69*, 46–61. JOUR.

Momoh, J. A., Guo, S. X., Ogbuobiri, E. C., & Adapa, R. (1994). The quadratic interior point method solving power system optimization problems. *IEEE Transactions on Power Systems*, *9*(3), 1327–1336. JOUR.

Naderi, E., Narimani, H., Fathi, M., & Narimani, M. R. (2017). A Novel Fuzzy Adaptive Configuration of Particle Swarm Optimization to Solve Large-Scale Optimal Reactive Power Dispatch. *Applied Soft Computing*. JOUR.

Nualhong, D., Chusanapiputt, S., Phomvuttisarn, S., & Jantarang, S. (2004). Reactive tabu search for optimal power flow under constrained emission dispatch. In *TENCON 2004.*

2004 IEEE Region 10 Conference (Vol. 100, pp. 327–330). CONF, IEEE.

Ozturk, A., Cobanli, S., Erdogmus, P., & Tosun, S. (2010). Reactive power optimization with artificial bee colony algorithm. *Scientific Research and Essays*, *5*(19), 2848–2857. JOUR.

Pan, W.-T. (2012). A new fruit fly optimization algorithm: taking the financial distress model as an example. *Knowledge-Based Systems*, *26*, 69–74. JOUR.

Parmar, S. A., Pandya, M. H., Bhoye, M., Trivedi, I. N., Jangir, P., & Ladumor, D. (2016). Optimal active and Reactive Power dispatch problem solution using Moth-Flame Optimizer algorithm. In *Energy Efficient Technologies for Sustainability (ICEETS), 2016 International Conference on* (pp. 491–496). CONF, IEEE.

Perez Abril, I., & Quintero, J. A. G. (2003). VAr compensation by sequential quadratic programming. *Power Systems, IEEE Transactions on*, *18*(1), 36–41. Journal Article. https://doi.org/10.1109/TPWRS.2002.807049

Price, K., Storn, R. M., & Lampinen, J. A. (2006). *Differential evolution: a practical approach to global optimization*. BOOK, Springer Science & Business Media.

Pudjianto, D., Ahmed, S., & Strbac, G. (2002). Allocation of VAR support using LP and NLP based optimal power flows. *IEE Proceedings-Generation, Transmission and Distribution*, *149*(4), 377–383. JOUR.

Radosavljević, J., Jevtić, M., & Milovanović, M. (n.d.). A solution to the ORPD problem and critical analysis of the results. *Electrical Engineering*, 1–13. JOUR.

Rajabioun, R. (2011). Cuckoo optimization algorithm. *Applied Soft Computing*, *11*(8), 5508–5518. JOUR.

Rajan, A., & Malakar, T. (2016). Exchange market algorithm based optimum reactive power dispatch. *Applied Soft Computing*, *43*, 320–336. JOUR.

Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2009). GSA: a gravitational search algorithm. *Information Sciences*, *179*(13), 2232–2248. JOUR.

Rashedi, E., Nezamabadi-Pour, H., & Saryazdi, S. (2010). BGSA: binary gravitational search algorithm. *Natural Computing*, *9*(3), 727–745. JOUR.

Ray, D., & Murillo-Sanchez, Z. C. E. (2015). Matpower 5.1 User's Manual. GEN, March.

Saadat, H. (1999). *Power system analysis*. BOOK, WCB/McGraw-Hill.

Sakr, W. S., EL-Sehiemy, R. A., & Azmy, A. M. (2017). Adaptive Differential Evolution Algorithm for Efficient Reactive Power Management. *Applied Soft Computing*. JOUR.

Shaw, B., Mukherjee, V., & Ghoshal, S. P. (2014). Solution of reactive power dispatch of power systems by an opposition-based gravitational search algorithm. *International Journal of Electrical Power & Energy Systems*, *55*, 29–40. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.ijepes.2013.08.010

Singh, R. P., Mukherjee, V., & Ghoshal, S. P. (2015). Optimal reactive power dispatch by particle swarm optimization with an aging leader and challengers. *Applied Soft Computing*, *29*, 298–309. JOUR.

Sinsuphan, N., Leeton, U., & Kulworawanichpong, T. (2013). Optimal power flow solution using improved harmony search method. *Applied Soft Computing*, *13*(5), 2364–2374. JOUR.

Storn, R., & Price, K. (1995). *Differential Evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces. International Computer Science Institute, Berkeley* (RPRT). CA, 1995, Tech. Rep. TR-95–012.

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of Global Optimization*, *11*(4), 341–359. JOUR.

Subbaraj, P., & Rajnarayanan, P. N. (2009). Optimal reactive power dispatch using self-adaptive real coded genetic algorithm. *Electric Power Systems Research*, *79*(2), 374–381. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.epsr.2008.07.008

Sulaiman, M. H., Mohamed, M. R., Mustaffa, Z., & Aliman, O. (2016). An Application of Multi-Verse Optimizer for Optimal Reactive Power Dispatch Problems. *International Journal of Simulation--Systems, Science & Technology*, *17*(41). JOUR.

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

Sulaiman, M. H., Mustaffa, Z., Daniyal, H., Mohamed, M. R., & Aliman, O. (2006). SOLVING OPTIMAL REACTIVE POWER PLANNING PROBLEM UTILIZING NATURE INSPIRED COMPUTING TECHNIQUES. Journal Article.

Sulaiman, M. H., Mustaffa, Z., Mohamed, M. R., & Aliman, O. (2015). Using the gray wolf optimizer for solving optimal reactive power dispatch problem. *Applied Soft Computing*, *32*, 286–292. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.asoc.2015.03.041

Talbi, E.-G. (2009). *Metaheuristics: from design to implementation* (Vol. 74). BOOK, John Wiley & Sons.

Trivedi, I. N., Bhesdadiya, R. H., Pandya, M. H., Jangir, N., Jangir, P., & Ladumor, D. (n.d.). Implementation of Meta-heuristic Levy Flight Moth-Flame Optimizer for Solving Real Challenging Constrained Engineering Optimization Problems. JOUR.

Trivedi, I. N., Jangir, P., Parmar, S. A., & Jangir, N. (n.d.). Optimal power flow with voltage stability improvement and loss reduction in power system using Moth-Flame Optimizer. *Neural Computing and Applications*, 1–16. JOUR.

Trivedi, I. N., Parmar, S. A., Pandya, M. H., Jangir, P., Ladumor, D., & Bhoye, M. T. (n.d.). Optimal Active and Reactive Power Dispatch Problem Solution using Moth-Flame Optimizer. JOUR.

Varadarajan, M., & Swarup, K. S. (2008a). Differential evolution approach for optimal reactive power dispatch. *Applied Soft Computing*, *8*(4), 1549–1561. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.asoc.2007.12.002

Varadarajan, M., & Swarup, K. S. (2008b). Differential evolutionary algorithm for optimal reactive power dispatch. *International Journal of Electrical Power & Energy Systems*, *30*(8), 435–441. Journal Article. https://doi.org/http://dx.doi.org/10.1016/j.ijepes.2008.03.003

Varadarajan, M., & Swarup, K. S. (2008c). Network loss minimization with voltage security using differential evolution. *Electric Power Systems Research*, *78*(5), 815–823. JOUR.

Vesterstrom, J., & Thomsen, R. (2004). A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In *Evolutionary Computation, 2004. CEC2004. Congress on* (Vol. 2, pp. 1980–1987). CONF, IEEE.

Vlachogiannis, J. G., & Lee, K. Y. (2006). A Comparative Study on Particle Swarm Optimization for Optimal Steady-State Performance of Power Systems. *Power Systems, IEEE Transactions on*, *21*(4), 1718–1728. Journal Article. https://doi.org/10.1109/TPWRS.2006.883687

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for optimization. *Evolutionary Computation, IEEE Transactions on*, *1*(1), 67–82. Journal Article.

Wu, Q. H., Cao, Y. J., & Wen, J. Y. (1998). Optimal reactive power dispatch using an adaptive genetic algorithm. *International Journal of Electrical Power & Energy Systems*, *20*(8), 563–569. JOUR.

Wu, Q. H., & Ma, J. T. (1994). Genetic search for optimal reactive power dispatch of power systems. In *Control, 1994. Control'94. International Conference on* (Vol. 1, pp. 717–722). CONF, IET.

Wu, Q. H., & Ma, J. T. (1995). Power system optimal reactive power dispatch using evolutionary programming. *IEEE Transactions on Power Systems, 10*(3), 1243–1249. JOUR.

Wu, Y.-C., Debs, A. S., & Marsten, R. E. (1994). A direct nonlinear predictor-corrector primal-dual interior point algorithm for optimal power flows. *IEEE Transactions on Power Systems*, *9*(2), 876–883. JOUR.

Yan, W., Liu, F., Chung, C. Y., & Wong, K. P. (2006). A hybrid genetic algorithm-interior point method for optimal reactive power flow. *IEEE Transactions on Power Systems*, *21*(3), 1163–1169. JOUR.

Yan, W., Lu, S., & Yu, D. C. (2004). A novel optimal reactive power dispatch method based on an improved hybrid evolutionary programming technique. *IEEE Transactions on Power Systems*, *19*(2), 913–918. JOUR.

Yang, X.-S. (2010a). A new metaheuristic bat-inspired algorithm. In *Nature inspired cooperative strategies for optimization (NICSO 2010)* (pp. 65–74). CHAP, Springer.

Yang, X.-S. (2010b). Firefly algorithm, stochastic test functions and design optimisation. *International Journal of Bio-Inspired Computation*, *2*(2), 78–84. JOUR.

Yang, X.-S. (2012). Flower pollination algorithm for global optimization. In *International Conference on Unconventional Computing and Natural Computation* (pp. 240–249). CONF, Springer.

Yang, X.-S., & Deb, S. (2009). Cuckoo search via Lévy flights. In *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on* (pp. 210–214). CONF, IEEE.

Yildiz, A. R. (2008). Hybrid Taguchi-harmony search algorithm for solving engineering optimization problems. *International Journal of Industrial Engineering: Theory, Applications and Practice*, *15*(3), 286–293. JOUR.

Yoshida, H., Kawata, K., Fukuyama, Y., Takayama, S., & Nakanishi, Y. (2000). A particle swarm optimization for reactive power and voltage control considering voltage security assessment. *IEEE Transactions on Power Systems*, *15*(4), 1232–1239. JOUR.

Zhang, J., & Sanderson, A. C. (2009). JADE: adaptive differential evolution with optional external archive. *IEEE Transactions on Evolutionary Computation*, *13*(5), 945–958. JOUR.

Zhang, W., & Liu, Y. (2008). Multi-objective reactive power and voltage control based on fuzzy optimization strategy and fuzzy adaptive particle swarm. *International Journal of Electrical Power & Energy Systems*, *30*(9), 525–532. JOUR.

Zhang, Y., & Ren, Z. (2004). Real-time optimal reactive power dispatch using multi-agent technique. *Electric Power Systems Research*, *69*(2), 259–265. JOUR.

Zhao, B., Guo, C. X., & Cao, Y. J. (2005). A multiagent-based particle swarm optimization approach for optimal reactive power dispatch. *Power Systems, IEEE Transactions on*, *20*(2), 1070–1078. Journal Article. https://doi.org/10.1109/TPWRS.2005.846064

Zimmerman, R. D., Murillo, S., x, nchez, C. E., & Thomas, R. J. (2011). MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education. *Power Systems, IEEE Transactions on*, *26*(1), 12–19. Journal Article. https://doi.org/10.1109/TPWRS.2010.2051168

APPENDIX A

LIST OF PUBLICATIONS

Rebecca Ng Shin Mei, Mohd Herwan Sulaiman**,** Zuriani Mustaffa, Hamdan Daniyal, *Optimal Reactive Power Dispatch Solution by Loss Minimization using Moth-Flame Optimization Technique*, Applied Soft Computing, *Volume 59*, *October 2017*, *Pages 210-222*. **ISI Impact factor in 2016: 3.541**

Sulaiman, M.H., Mustaffa, Z., Mohamed, M.R., Aliman, O., An application of multi-verse optimizer for optimal reactive power dispatch problems, International Journal of Simulation: Systems, Science and Technology, Volume 17, Issue 41, 2017, Pages 5.1-5.5

Rebecca Ng Shin Mei, Mohd Herwan Sulaiman, Hamdan Daniyal and Zuriani Mustaffa, International Conference on Electrical Control and Computer Engineering 2017 (INECCE 2017), Langkawi, 16-17 October 2017.

Mohd Herwan Sulaiman**,** Zuriani Mustaffa, *Cuckoo Search Algorithm as an Optimizer for Optimal Reactive Power Dispatch Problems*, 3rd International Conference on Control, Automation and Robotics (ICCAR2017), Nagoya, Japan, 24-26 April 2017.

Rebecca Ng Shin Mei, Mohd Herwan Sulaiman, Zuriani Mustaffa, *Ant Lion Optimizer for Optimal Reactive Power Dispatch Solution*, Journal of Electrical Systems "Special Issue AMPE2015", pp. 68-74. **Abs./Ind.: SCOPUS**.

Mohd Herwan Sulaiman**,** Zuriani Mustaffa, Omar Aliman, Hamdan Daniyal and Mohd Rusllim Mohamed, *An Application of Moth-Flame Optimization Algorithm for Solving Optimal Reactive Power Dispatch*, 4th IET International Conference on Clean Energy and Technology (CEAT 2016), 14-15 Nov 206, K. Lumpur , Malaysia.

Applied Soft Computing 59 (2017) 210–222

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Optimal reactive power dispatch solution by loss minimization using moth-flame optimization technique

Rebecca Ng Shin Mei [a,*], Mohd Herwan Sulaiman [a], Zuriani Mustaffa [b], Hamdan Daniyal [a]

[a] Faculty of Electrical & Electronics Engineering, University Malaysia Pahang (UMP), 26600 Pekan, Pahang, Malaysia
[b] Faculty of Computer Systems & Software Engineering, University Malaysia Pahang (UMP), 26300 Gambang, Pahang, Malaysia

ARTICLE INFO

Article history:
Received 29 April 2016
Received in revised form 6 April 2017
Accepted 30 May 2017

Keywords:
Moth-flame optimization algorithm
Nature-inspired heuristic technique
Optimal reactive power dispatch
Loss minimization
Voltage deviation minimization

ABSTRACT

In this paper, a newly surfaced nature-inspired optimization technique called moth-flame optimization (MFO) algorithm is utilized to address the optimal reactive power dispatch (ORPD) problem. MFO algorithm is inspired by the natural navigation technique of moths when they travel at night, where they use visible light sources as guidance. In this paper, MFO is realized in ORPD problem to investigate the best combination of control variables including generators voltage, transformers tap setting as well as reactive compensators sizing to achieve minimum total power loss and minimum voltage deviation. Furthermore, the effectiveness of MFO algorithm is compared with other identified optimization techniques on three case studies, namely IEEE 30-bus system, IEEE 57-bus system and IEEE 118-bus system. The statistical analysis of this research illustrated that MFO is able to produce competitive results by yielding lower power loss and lower voltage deviation than the selected techniques from literature.

1. Introduction

Over the last few decades, electrical power system has become an increasingly important subject due to the modern economy that run by electricity. Electrical power system is a system of generating, transmitting and distributing electricity for industrial, housing and transportation uses. Moreover, electrical power system is also the heart of renewable energy systems. As the demands for electricity increased, the consumption of resources also will gradually increase. Undeniably, optimal reactive power dispatch (ORPD) plays an important role in operation and control of power system due to its remarkable influence on the reliability, security and economic operation issues. As a sub problem of optimal power flow (OPF), ORPD is defined as a renowned nonlinear optimization problem in power system which involving both discrete and continuous control variables while satisfying both equality and inequality constraints [1–5]. Thence, optimization process is utilized to obtain the best possible combinational of control variables including generator bus voltages, transformers tap setting and reactive compensators sizing in order to minimize the objective functions.

There is variety of optimization techniques in overcoming ORPD problem as reported in literature. Referring to [6–11], conventional optimization methods such as linear programming [12,13], non-linear programming, quadratic programming [14], Newton method, gradient-based algorithm and interior point method [15] have been implemented in solving ORPD problem. Nevertheless, they are inefficient in dealing problems with nonlinear functions and discrete variables [1,16], thus, leading loss of accuracy. Furthermore, the so-called stochastic search optimization methods such as genetic algorithm (GA) [17,18], evolutionary programming (EP) [19], evolutionary strategy (ES) and tabu search (TS) were also applied to overcome the ORPD problem. The key success of stochastic search methods are their ability in obtaining global optimum and handling non-convex as well as discontinuous objective functions. However, they are inefficient in managing problems with discrete nature and integer [20].

Recently, development and exploitation in meta-heuristic methods have shown a better result in solving ORPD problem. Those methods include particle swarm optimization (PSO) [1,21], artificial bee colony (ABC) [20], harmony search algorithm (HSA) [3], improved HSA (IHSA) [22], modified HSA [23], gravitational search algorithm (GSA) [24,25], seeker optimization algorithm (SOA) [26] and gray wolf optimizer (GWO) [2]. Additionally, there are also researchers who used hybrid techniques to solve ORPD problem such as combined modified imperialist competitive algorithm and invasive weed optimization (MICA-IWO) [4], combined differential evolution and ant system [27] as well as hybrid particle swarm optimization and imperialist competitive algorithm (PSO-ICA) [28]. In [29], quasi-oppositional differential evolution (QODE) is proposed to solve ORPD problem by implementing quasi-oppositional based learning (QOBL). In [30], the researchers

* Corresponding author.
E-mail address: rebeccamei527@gmail.com (R. Ng Shin Mei).

http://dx.doi.org/10.1016/j.asoc.2017.05.057
1568-4946/

proposed a two-point estimate method (TPEM) to model the load uncertainty in a multi objective ORPD (MO-ORPD) problem. However, according to no free lunch (NFL) theorem [31], there is no specific technique that can solve all the optimization problem. Therefore, ORPD problem still can be solved by implementing new developed optimization algorithm.

This paper proposes the use of a novel nature-inspired heuristic technique known as moth-flame optimization (MFO) algorithm in solving ORPD problem. This technique serves as an alternative to other recent optimization techniques. The MFO algorithm has been developed by Seyedali Mirjalili [32], which inspired by the nature navigation method of moths at dark by travelling depending on a light source. In comparison with other methods, there are several contributions of this algorithm in solving optimization problem. First, MFO algorithm implements a population of moths to perform optimization and each moth is required to update their positions with respect to a flame. Thus, this helps to avoid the local optima entrapment and improve the exploration process in the search space. The moths will always update their positions according to the most promising flames obtained so far over the course of iteration. The flames will be remembered as the best optimum solutions. These serve as guidance for the moths, thence, this help them to retain the best results. Consequently, the convergence of MFO is ensured. Moreover, MFO algorithm is simple to implement as it does not required many control parameters while solving ORPD problem.

The rest of this paper is organized as follows: Section 2 discusses the mathematical formulation of ORPD problem followed by a concise introduction of MFO algorithm in Section 3. Section 4 presents the utility of MFO by implementing this algorithm in solving ORPD problem. The simulation results and discussions are provided in Section 5. Last but not least, Section 6 concludes the research of this paper.

2. ORPD mathematical formulation

2.1. Objective function

In this paper, the objective functions of ORPD problem are to minimize power losses and voltage deviation of the transmission system while fulfilling the equality constraint and inequality constraints. The ORPD problem can be formulated as the minimization of function $f(x, u)$ as described as follows:

Minimize $f(x, u)$

$$\text{Subjected to} \begin{array}{l} g(x,u)=0 \\ h(x,u)\leq 0 \end{array} \tag{1}$$

where function $f(x, u)$ is the objective function. Additionally, $g(x, u) = 0$ and $h(x, u) \leq 0$ are the equality and inequality constraints respectively. In ORPD, the equality constraint is the power balanced equation whereas the inequality constraints are generators voltage, transformers tap setting and reactive compensators sizing. x and u are the dependent variables vector and control variables vector respectively. As mentioned before, one of the objective functions of this paper is to minimize the total system transmission loss, which it is in fact an economic loss that neither provides any benefit nor profit. The other objective function, minimizing voltage deviation is important as well as it increase overall system stability. The total system transmission loss, F_1 and voltage deviation at load buses, F_2 can be formulated as follows [3]:

$$F_1 = P_{Loss}(x,u) = \sum_{L=1}^{Nl} P_{Loss} \tag{2}$$

$$F_2 = VD(x,u) = \sum_{i=1}^{Nd} |V_i - V_i^{sp}| \tag{3}$$

where Nl indicates the number of transmission lines and Nd is the number of load buses. V_i is the voltage at load bus-i and V_i^{sp} is the specified value (usually set as 1.0 p.u).

2.2. Equality constraint

The equality constraint which is the power equality of load flow stated that the difference between power generated and power demand is equal to power loss as declared in [3]. The equality constraint equations can be expressed as below:

$$P_{Gi} - P_{Di} = V_i \sum_{j \in N_i} V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij}\right) \tag{4}$$

$$Q_{Gi} - Q_{Di} = V_i \sum_{j \in N_i} V_j \left(B_{ij} \cos\theta_{ij} - G_{ij} \sin\theta_{ij}\right) \tag{5}$$

where V_i and V_j are the voltage at load bus-i and bus-j respectively, B_{ij} and G_{ij} are the susceptance and conductance between bus-i and bus-j respectively. On the other hand, P_{Gi} and P_{Di} are the real power generation and real load demand respectively. Whereas, Q_{Gi} and Q_{Di} are the reactive power generation and reactive load demand respectively.

2.3. Inequality constraints

2.3.1. Generator constraints: bus voltages' generation as well as generation of real and reactive power must be restricted by their boundaries as below

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

$$Q_{Gi}^{\min} \leq Q_{Gi} \leq Q_{Gi}^{\max} \quad i = 1, \ldots, N_G \tag{7}$$

$$V_{Gi}^{\min} \leq V_{Gi} \leq V_{Gi}^{\max} \quad i = 1, \ldots, N_G \tag{8}$$

where N_G is the generators' number.

2.3.2. Transformer tap ratios must be within their minimum and maximum boundaries as below

$$T_i^{\min} \leq T_i \leq T_i^{\max} \quad i = 1, \ldots, N_T \tag{9}$$

where N_T is the transformers' number.

2.3.3. Reactive compensator sizes are limited by their ranges as below

$$Q_{Ci}^{\min} \leq Q_{Ci} \leq Q_{Ci}^{\max} \quad i = 1, \ldots, N_C \tag{10}$$

where N_C is the reactive compensators' number.

In this paper, it is worth noting that a special tool has been applied which is the MATPOWER software package [33] in order to achieve the objective functions. This package is utilized to ensure precise results can be attained by running the load flow program.

3. Moth-flame optimizer (MFO)

Moth-flame optimization (MFO) algorithm was initially developed by Seyedali Mirjalili [32] and being proven to be competitive with other well-known optimization techniques. In nature, moths are insects that are highly close to the butterflies' family. During their lifetime, they basically undergo two main milestones which are larvae stage before evolve to adult stage. The inspiration of this algorithm is the unique navigation technique of moths during night time. The moths used a mechanism known as transverse orientation when travel in dark which depending on the moonlight. They travelled by retaining their position at a fixed angle with

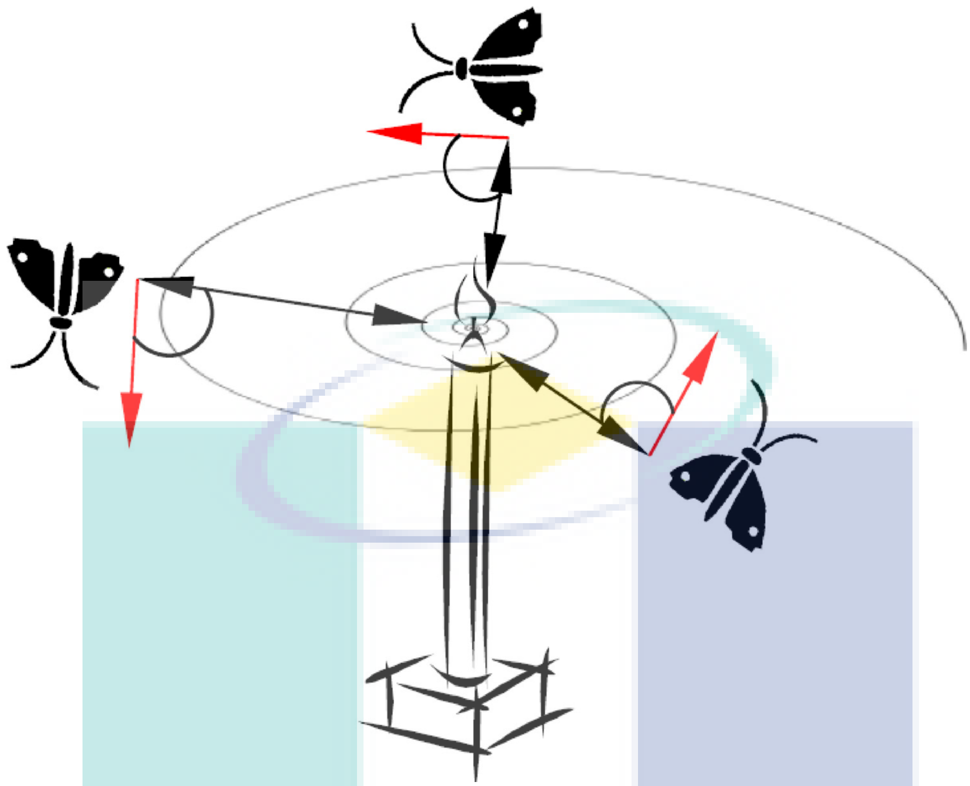

Fig. 1. Moth's spiral flying path around close light source [32].

respect to the light source. In fact, the moths are mostly attracted by man-made light source such as the lamp and flied around the light in spiral shape. Additionally, they also tried to retain the similar angle with respect to the man-made light source. However, this behaviour causes a deadly spiral fly path for them as the light source is much closer as compared with the moon [32,34]. This natural behaviour of moth's spiral flying path is illustrated in Fig. 1.

3.1. MFO mathematical formulation

As to model the MFO algorithm, the first key component is the set of moths, which can be expressed in matrix as follows:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & m_{n,d} \end{bmatrix} \tag{11}$$

where d indicates the dimension (number of variables) and n is the number of moths. It is supposed that moths are the candidate solutions and the position of moths in the space is the problem's variables. The second key component is the set of flames which can be expressed in matrix similar to the matrix of moth, M as follows:

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & F_{n,d} \end{bmatrix} \tag{12}$$

where d and n are the dimension as well as the number of moths respectively. As the dimension of Eqs. (11) and (12) are the same, it is assumed that there is an array which storing the corresponding fitness values as follows:

$$OM = \begin{bmatrix} OM_1 \\ \vdots \\ \vdots \\ OM_n \end{bmatrix} \tag{13}$$

$$OF = \begin{bmatrix} OF_1 \\ \vdots \\ \vdots \\ OF_n \end{bmatrix} \tag{14}$$

where n is the number of moths. It is vital to notice that moth and flame both are solutions. However, the difference between them is the way to treat and update them. In MFO, flames are the best position of moths acquires so far and moths are the actual search agents that move around the search space. Therefore, flames are the pins that are dropped by the moths when searching the search space. Each moth seeks around a flame and updates its position in order to find a better result. This mechanism helps the moth not to lose its best result [32]. As mentioned before, the position of each moth is updated depending to a flame which can be mathematically modeled by the equation below [32]:

$$M_i = S\left(M_i, F_j\right) \tag{15}$$

where M_i and F_j represent the i-th moth and j-th flame respectively whereas S is the spiral function. The following expression is the logarithmic spiral function which is the main update mechanism of moth [32]:

$$S\left(M_i, F_j\right) = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \tag{16}$$

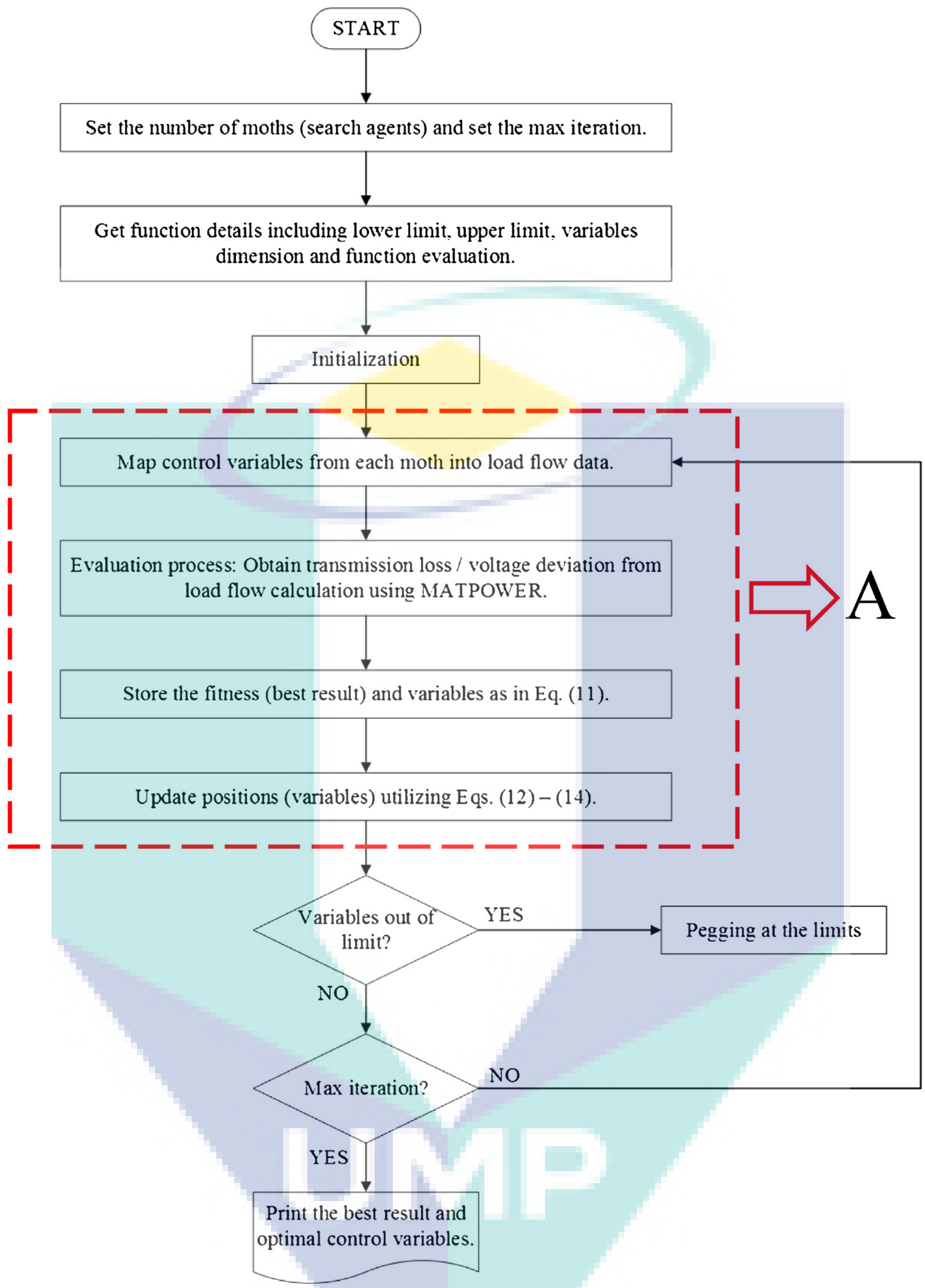

Fig. 2. Flowchart of MFO algorithm for solving ORPD problem.

where b indicates a constant for defining the shape of the logarithmic spiral, t is a random number within the interval of [−1,1] and D_i represents the distance of the i-th moth for the j-th flame where it can be calculated as below:

$$D_i = |F_j - M_i| \tag{17}$$

where D_i is the distance of the i-th moth for the j-th flame, F_j represents the j-th flame and M_i represents the i-th moth. Since Eq. (16) permits a moth to fly spirally around a flame and not essentially within space between them, this guarantees exploration and exploitation processes of the search area can be achieved. Exploration happens when the next position lies outside the area between the flame and the moth. On the other hand, exploitation occurs when the next position situated within the area between the flame and the moth. In order to avoid being trapped in local optima, each moth is forced to update its position utilizing one of the flames in Eq. (16). The flames are sorted and classified depending on their fitness values in each iteration. Then, the moths updated their positions based upon their corresponding flames [32].

It is important to note that before the beginning of the first iteration, there is N number of flames. However, in the final stages of the iterations, the moths update their positions only based upon the best flame. Thus, the number of flames will decreases gradually with iterations. This decrement in number of flames balances the exploitation and exploration of the search area [32]. The following equation is expressed for the number of flames regarding this phenomenon:

$$flame\ no = round\left(N - l * \frac{N-1}{T}\right) \quad (18)$$

where N is the maximum number of flames, l indicates the current number of iteration and T is the maximum iterations.

4. Implementing MFO in ORPD problem

The application of MFO algorithm in overcoming ORPD problem is via the searching of the optimal values of control variables to minimize the objective functions while satisfying the equality constraint and inequality constraints. Firstly, the maximum iteration and the number of search agents (number of moths) are set. MFO algorithm is a population based algorithm. The candidate solution (population) is expressed in matrix form as described in Eq. (11). The column of the matrix is the number of control variables and the row is the number of search agents.

values in matrix form. The best solution obtained will be situated at the upper part of the matrix while the worst solution will be situated at the lower part of the matrix. Then, the updated control variables will be checked whether they are out of limits from the constraints. If the control variables are out of limit, they will be tagged at the lower and upper limits in order to obtain accurate results. The simulation continues by evaluating the MFO processes (Eqs. (11)–(18)). The evaluation process repeated until it is terminated by the maximum iteration limit. For the second objective function, the same MFO step is utilized to optimize the voltage deviation at load buses. Additionally, the voltage magnitude of each load bus must be in the range of ±10%. The implementation of MFO algorithm in overcoming ORPD problem is illustrated in Fig. 2.

The main evaluation process of MFO algorithm in solving ORPD problem is depicted in part A (red dashed rectangle) as shown in Fig. 2. The optimization process of MFO algorithm is further explained by the pseudo code below:

```
begin
initialize the positions of moths
while ( iteration < max iteration )
update the number of flames according to Eq.18
for ( i=1:size (moth_pos,1) )      //moth_pos: position of moth
check if moths go out of the search space
if the moth is out of the search space, bring it back
calculate the fitness of the moths
end
if ( iteration==1 )
sort the first population of the moths
update the flames
else
sort the moths
update the flames
end
update the position of the best flame obtained so far
for ( i=1:size (moth_pos,1) )
    for ( j=1:size (moth_pos,2) )
        if  ( i <= flame_no )      //flame_no: number of flame
        update the position of the moth with respect to its corresponding flame using Eq.15
        end
        if ( i > flame_no )
        update the position of the moth with respect to only one flame
        end
    end
end
print the best optimum results obtained so far
end
```

During evaluation process, the position of each moth is mapped into the load flow data. Then, the load flow program is executed to obtain the transmission loss. In each iteration, each moth updates its position with respect to the flame such in Eqs. (12)-(14). After updating the positions, the transmission loss is obtained for the corresponding moth. The solution is sorted based upon their fitness

5. Simulations and discussions

In order to prove the effectiveness of MFO algorithm in solving ORPD problem, MATLAB on Window 7 Professional Intel® Core™ i3-2330M CPU @ 2.2 GHz 6GB RAM is utilized to run the simulations. In this paper, three case studies including IEEE 30-bus system, IEEE

Table 1
Boundary setting of control variables for IEEE 30-bus system [2,3].

Control Variables	Lower Bound	Upper Bound
Generator Voltages	0.9 p.u	1.1 p.u
Transformer Tap Setting	0.95 p.u	1.05 p.u
Reactive Compensator Sizing	−12 MVar	36 MVar

57-bus system and IEEE 118-bus system are carried out to prove the efficiency of MFO algorithm. Moreover, their statistical results are compared against the selected optimization algorithms from literature.

5.1. IEEE 30-bus system with 13 control variables

Initially, case study of IEEE 30-bus system will be tested. It is designed based on [2] and [3]. This case study constitutes 13 control variables which need to be optimized. The IEEE 30-bus system comprised of six generators, 41 lines and four transformers. The transformers are situated at lines 6-9, 6-10, 4-12 and 27-28. Additionally, three reactive compensation elements are located at buses 3, 10 and 24. The limit setting for the control variables including generator voltages, transformer tap ratios and reactive compensators sizing are set in a range of ±10% as depicted in Table 1 in order to satisfy the equality and inequality constraints. Moreover, the load demand for this system is set as $S = P + jQ = 2.832 + j1.262\ p.u.$

For this case study, the results reported in [2–4] are mapped into the same MATPOWER load flow program to evaluate the total transmission loss as for fair comparison with other selected optimization algorithms. Table 2 illustrates the results of control variables and the total power losses. It can be observed that MFO algorithm performs better than other optimization algorithms. In this table, two best results of MFO utilizing 20 and 30 search agents depicted as MFO#2 and MFO#3 respectively are tabulated. It is noted that 30 search agents is able to obtain better result than 20 search agents.

Furthermore, the simulation results of MFO are compared with well-known methods in order to expose the superiority of MFO algorithm. These included particle swarm optimization (PSO) [3], differential evolution (DE) [35] and genetic algorithm (GA) [36]. All the initial settings and constraints of the selected methods are set according to the settings used by MFO algorithm. This is to coincide ORPD problem solved by DE method and GA method with MFO algorithm and obtain a reasonable comparison. The comparison between MFO algorithm and the selected methods (PSO, DE and GA) provides 22.02%, 6.17% and 5.97% reduction of transmission losses respectively utilizing 30 search agents. This achievement indicates that MFO algorithm leads to a better result than the well-recognized methods in solving ORPD problem.

Additionally, the simulation results obtained by MFO algorithm for power loss reduction are also compared with other recently developed algorithms namely MICA-IWO [4], harmony search algorithm (HSA) [3] and gray wolf optimizer (GWO) [2]. Again, the comparison between MFO with MICA-IWO, HSA and GWO gives 5.35%, 10.23% and 0.26% reduction of power loss respectively utilizing 30 search agents. Other algorithms reported in [3] and [4] including invasive weed optimization (IWO), imperialist competitive algorithm (ICA) and simple genetic algorithm (SGA) are also presented in Table 2. Judging from the tabulated results, it is concluded that MFO algorithm has the best performance by producing the lowest total power loss. It is robust among all rival approaches.

Table 3 shows the minimum value of power loss in MW and their percentage of loss reduction obtained by different techniques. The total power loss is reduced to a minimum value of 4.5865 MW (best case) from the base case loss of 5.663 MW by using MFO algorithm. It is about 19.01% of loss reduction. From this achievement, MFO algorithm is able to excel GWO algorithm. GWO algorithm produces about 18.80% of loss reduction from the base case loss of 5.663 MW. Additionally, it can be noticed that the obtained control variables as shown in Table 2 are all within the range of the limit as stated in Table 1. The number of function evaluation (NFE) to reach the best results in this case study is 4500.

Fig. 3 illustrates the performance of MFO in term of loss (MW) versus iteration where the maximum iteration is set as 150 for four different number of search agents. It is vital to emphasize that the results plotted in the figure are the best results for 30 free running simulations that have been chosen. Furthermore, it can be observed that 30 search agents is able to get merely better result than 20 search agents. The power losses for all different search agents reduce over the iterations and converge to a minimum value. From this figure, it can be concluded that as the search agent number increased, a better convergence to the optimum result can be achieved. However, 30 search agents show the best convergence among other number of search agents including 40 search agents. Thence, 30 search agents was used for all the case studies in this paper.

Figs. 4 and 5 show the performance of MFO for 30 search agents and 20 search agents respectively for 30 free running of simulations. The best, worst and average results utilizing 30 search agents are 4.5864 MW, 4.7486 MW and 4.6081 MW. On the other hand, the best, worst and average results using 20 search agents are 4.5867 MW, 5.2313 MW and 4.67494 MW. According to Figs. 4 and 5, it can be concluded that 30 search agents can obtain better solutions for the best, average and worst results as compared to 20 search agents. Thence, by using 30 search agents, MFO algorithm can achieved better solution with smaller range of power losses. In addition, the comparison between the performances of 20 and 30 search agents are plotted in Fig. 6. It is worth to highlight that 30 search agents is adequate to obtain consistent results than 20 search agents. The range of the results using 30 search agents is

Table 2
ORPD results of control variables after optimization by MFO and other selected optimization algorithms for IEEE 30-bus system with 13 control variables.

Control Variables	MICA-IWO [4]	IWO [4]	ICA [4]	SGA [3]	HSA [3]	PSO [3]	DE [35]	GA [36]	GWO [2]	MFO#2	MFO#3
V_1	1.07972	1.06965	1.07850	1.0512	1.0726	1.0313	1.095318821	1.0721	1.100000	1.1000	1.1000
V_2	1.07055	1.06038	1.06943	1.0421	1.0625	1.0114	1.085946171	1.0630	1.096149	1.0943	1.0946
V_5	1.04836	1.03692	1.04698	1.0322	1.0399	1.0221	1.062627614	1.0377	1.080036	1.0752	1.0756
V_8	1.04865	1.03864	1.04714	0.9815	1.0422	1.0031	1.065076469	1.0445	1.080444	1.0770	1.0772
V_{11}	1.07518	1.02973	1.03485	0.9766	1.0318	0.9744	1.026600279	1.0132	1.093452	1.0696	1.0868
V_{13}	1.07072	1.05574	1.07106	1.1000	1.0681	0.9987	1.014253253	1.0898	1.100000	1.1000	1.1000
T_{6-9}	1.03	1.05	1.08	0.95	1.01	0.97	1.0177960	1.0221	1.04	1.05000	1.04110
T_{6-10}	0.99	0.96	0.95	0.98	1.00	1.02	0.9792765	0.9917	0.95	0.95000	0.95007
T_{4-12}	1.00	0.97	1.00	1.04	0.99	1.01	0.9778431	0.9964	0.95	0.95491	0.95541
T_{27-28}	0.98	0.97	0.97	1.02	0.97	0.99	1.0089383	0.9710	0.95	0.95783	0.95754
Q_{C3}	−7	8	−6	12	34	17	20.22358622	5.35020	12	7.0538	7.1032
Q_{C10}	23	35	36	−10	12	13	9.584327308	36.0000	30	36.000	30.796
Q_{C24}	12	11	11	30	10	23	13.02992421	12.4175	8	9.8889	9.8981
P_{Loss} (MW)	4.846	4.92	4.849	6.5318	5.109	5.8815	4.888080765	4.8775	4.5984	4.5867	4.5865

Table 3
Percentage of loss reduction after optimization by MICA-IWO, HSA, DE, GA, GWO and MFO for IEEE 30-bus system.

Compared item	Base case	MICA-IWO	HSA	DE	GA	GWO	MFO
P_{Loss} (MW)	5.663	4.846	5.109	4.888080765	4.8775	4.5984	4.5865
Percentage of loss reduction (%)	–	14.43	9.78	13.68	13.87	18.80	19.01

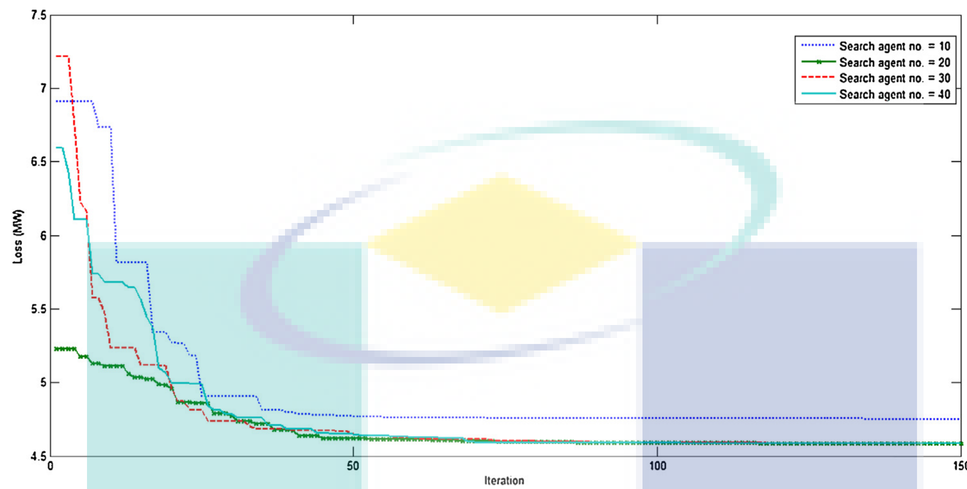

Fig. 3. Performance for different number of search agents using MFO algorithm (IEEE 30-bus system with 13 control variables).

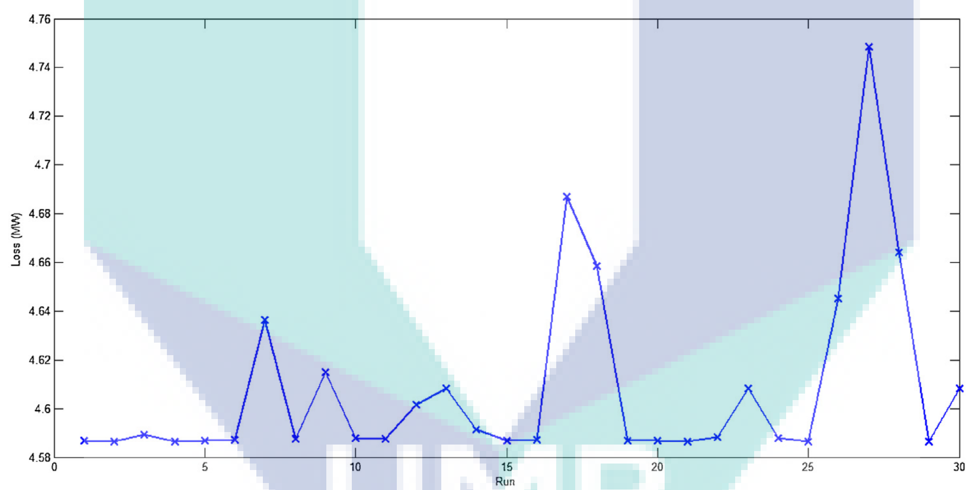

Fig. 4. Performance of 30 search agents for 30 free running simulations (IEEE 30-bus system with 13 control variables).

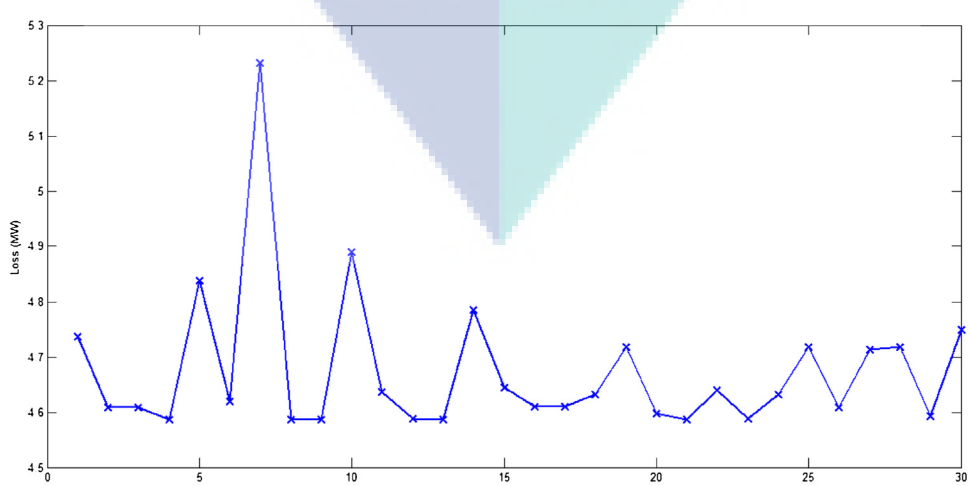

Fig. 5. Performance of 20 search agents for 30 free running simulations (IEEE 30-bus system with 13 control variables).

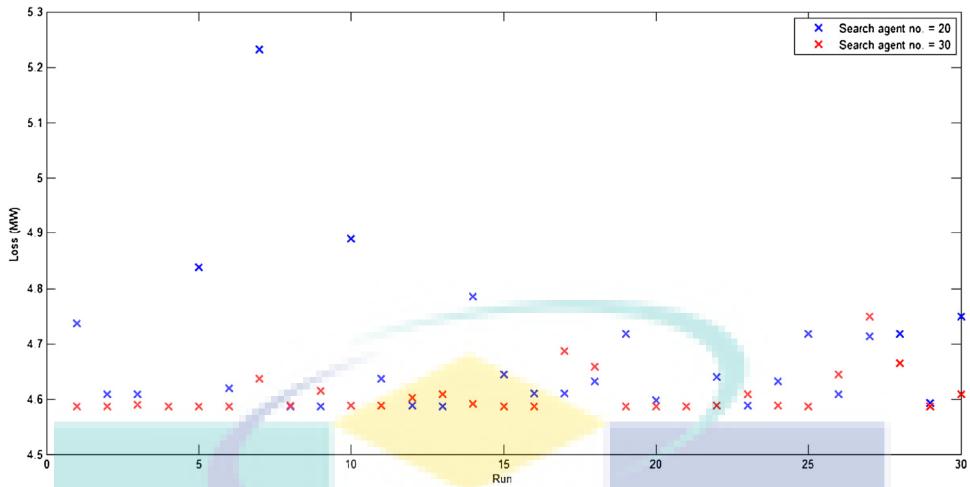

Fig. 6. Comparison between 20 and 30 search agents for 30 free running simulations (IEEE 30-bus system with 13 control variables).

Table 4
Comparison of voltage deviation (p.u) for different techniques on IEEE 30-bus system.

Compared item	SGA [3]	PSO [3]	HSA [3]	GWO [2]	MFO
Best deviation (p.u)	0.1501	0.1424	0.1349	0.12604	0.12154
Average deviation (p.u)	0.1523	0.1496	0.1443	0.14484	0.132961
Worst deviation (p.u)	0.1717	0.1639	0.1589	0.17273	0.15612

within 4.58 MW and 4.76 MW while the range of the results utilizing 20 search agents is within 4.5 MW and 5.3 MW.

The results of voltage deviation as second objective function are tabulated in Table 4. The performance of MFO for 30 running simulations is compared with the results obtained by simple genetic algorithm (SGA) [3], particle swarm optimization (PSO) [3], harmony search algorithm (HSA) [3] and gray wolf optimizer (GWO) [2]. From the simulation results, it can be concluded that MFO is able to decrease the voltage deviation more than the other optimization algorithms.

5.2. IEEE 30-bus system with 19 control variables

For this case study, there are 19 control variables need to be optimized which including six generator voltages, four transformers and nine compensator devices. Both of the real and reactive load demands for this case study are same as the previous case study. However, the limit settings for the control variables are different to the previous case study and they are set according to [37].

Table 5 illustrates the results obtained utilizing MFO and DE [37]. It is important to notice that the power loss tabulated in Table 5 is different with the power loss reported in [37] for the same parameters. All the results of power losses in Table 5 are obtained through MATPOWER load flow program. For fair comparison, the results of control variables obtained by DE [37] are mapped into the same MATPOWER load flow program to obtain the total power transmission loss (4.5179 MW). Judging from the table, it can be observed that the total power loss obtained by MFO is lower than the one obtained using DE. It is about 0.11% of reduction loss for the comparison between MFO and DE methods. Additionally, it is worth to note that all the obtained results for the control variables are converged within their limit range. The number of function evaluation (NFE) to reach the best solutions in this case study is same as the previous case study.

Table 5
ORPD results of control variables after optimization for IEEE 30-bus system with 19 control variables.

Control Variables	DE [37]	MFO
V_1	1.1	1.1
V_2	1.0931	1.0943
V_5	1.0736	1.0747
V_8	1.0756	1.0766
V_{11}	1.1	1.1
V_{13}	1.1	1.1
T_{11}	1.0465	1.0433
T_{12}	0.9097	0.9
T_{15}	0.9867	0.97912
T_{36}	0.9689	0.96474
Q_{C10}	5	5
Q_{C12}	5	5
Q_{C15}	5	4.8055
Q_{C17}	5	5
Q_{C20}	4.406	4.0263
Q_{C21}	5	5
Q_{C23}	2.8004	2.5193
Q_{C24}	5	5
Q_{C29}	2.5979	2.1925
P_{Loss} (MW)	4.5179	4.5128

5.3. IEEE 57-bus system

In order to further prove the ability of MFO algorithm in solving ORPD problem, case study of IEEE 57-bus system is utilized and the results obtained are compared with other optimization techniques. IEEE 57-bus system comprises of 25 control variables including seven generators, 15 transformers and three reactive compensation devices as referring to [22]. Additionally, the upper and lower limits of control variables that need to be optimized are tabulated in Table 6. The load demand for real and reactive power of this system are set as 1250.8 MW and 336.4 MVar respectively.

For this case study, the number of moths (search agents) and maximum iteration are set as 30 and 300 respectively. For fair comparison, the results reported in [38] and [39] are mapped into

Table 6
Boundary setting of control variables for IEEE 57-bus system [38,39].

Control Variables	Lower Bound	Upper Bound
Generator Voltages	0.94 p.u	1.06 p.u
Transformer Tap Setting	0.9 p.u	1.1 p.u
Q_{C18}	0 MVar	10 MVar
Q_{C25}	0 MVar	5.9 MVar
Q_{C53}	0 MVar	6.3 MVar

Table 7
ORPD results of control variables after optimization for IEEE 57-bus system.

Control Variables	BA	FA [39]	FPA	CSA [38]	GSA [38]	PSO [38]	DE [35]	GA [36]	GWO [39]	MFO
V_1	1.0603	1.0600	1.0599	1.0600	1.0600	1.0600	1.054927568	1.0600	1.0600	1.06000
V_2	1.0558	1.0572	1.0561	1.0582	1.0582	1.0600	1.047453328	1.0556	1.0562	1.05870
V_3	1.0456	1.0428	1.0472	1.0466	1.0462	1.0483	1.019982357	1.0320	1.0370	1.04690
V_6	1.0369	1.0366	1.0401	1.0409	1.0391	1.0423	1.000355275	1.0187	1.0202	1.04210
V_8	1.0499	1.0541	1.0585	1.0587	1.0600	1.0600	1.017291262	1.0380	1.0449	1.06000
V_9	1.0405	1.0355	1.0429	1.0417	1.0432	1.0432	1.010825629	1.0257	1.0294	1.04230
V_{12}	1.0314	1.0320	1.0387	1.0377	1.0379	1.0387	1.018695353	1.0258	1.0319	1.03730
T_{4-18}	0.9810	0.9312	0.9834	0.9440	0.9054	0.9000	0.946508732	1.0330	0.9847	0.95011
T_{4-18}	0.9921	0.9901	1.0559	1.0182	0.9978	1.1000	1.022067545	0.9056	0.9326	1.00760
T_{21-20}	1.0155	0.9845	1.0308	1.0207	1.0021	1.0314	1.007878565	0.9830	0.9576	1.00630
T_{24-26}	0.9962	1.0112	1.0620	1.0110	1.0180	1.0097	0.969228534	1.0028	0.9968	1.00760
T_{7-29}	0.9624	0.9683	1.0342	0.9744	0.9712	0.9754	0.945393886	0.9634	0.9636	0.97523
T_{34-32}	0.9520	0.9657	1.0160	0.9721	0.9692	0.9746	1.003513104	0.9835	0.9812	0.97218
T_{11-41}	0.8857	0.9762	0.9482	0.9015	0.9683	0.9000	1.075760701	0.9346	1.0621	0.90000
T_{15-45}	0.9736	0.9653	0.9743	0.9723	0.9717	0.9726	0.993151298	0.9669	0.9755	0.97186
T_{14-46}	0.9747	0.9524	0.9478	0.9537	0.9530	0.9538	0.972732845	0.9493	0.9639	0.95355
T_{10-51}	0.9550	0.9671	0.9663	0.9664	0.9691	0.9680	0.993603333	0.9632	0.9723	0.96736
T_{13-49}	0.9271	0.9291	0.9448	0.9269	0.9242	0.9264	0.957709946	0.9265	0.9248	0.92788
T_{11-43}	1.0396	1.0020	0.9856	0.9645	1.0387	1.1000	1.009537100	0.9605	0.9554	0.96406
T_{40-56}	1.0800	1.0224	1.0715	0.9943	1.0497	1.0624	1.010801305	1.0381	1.1000	0.99980
T_{39-57}	0.9838	1.0232	0.9783	0.9737	1.0668	1.0265	1.032377919	0.9815	0.9976	0.96060
T_{9-55}	1.0156	0.9687	1.0379	0.9750	0.9807	0.9764	0.977294007	0.9682	0.9845	0.97899
Q_{C18}	5.6561	4.1934	6.9194	9.2807	0.1863	9.9988	3.549722704	6.6369	1.8917	9.99680
Q_{C25}	2.4993	4.2297	3.7103	5.8943	4.0488	5.9000	3.435785358	5.8568	5.2489	5.90000
Q_{C53}	3.0430	5.9252	4.2180	6.2885	4.8099	6.3000	1.772303388	5.9162	5.1513	6.30000
P_{Loss} (MW)	24.9254	24.4587	24.8419	24.2619	24.4922	24.3826	25.95556947	24.5968	24.7523	24.25293

Table 8
Percentage of loss reduction after optimization by CSA, FA, GWO and MFO for IEEE 57-bus system.

Compared item	Base case	CSA	FA	GWO	MFO
P_{Loss} (MW)	27.864	24.2619	24.4587	24.7523	24.25293
Percentage of loss reduction (%)	–	12.93	12.22	11.17	12.96

the MATPOWER load flow program to obtain the total power loss. The results for the best combination of control variables and total power loss are presented in Table 7. The simulation results of MFO algorithm are compared with renowned optimization techniques including particle swarm optimization (PSO) [38], differential evolution (DE) [35] and genetic algorithm (GA) [36]. In order to conform the comparison, DE and GA methods are set with the same search agent number and iteration number which are 30 and 300 respectively. The comparison between MFO with PSO, DE and GA gives about 0.53%, 6.56% and 1.40% reduction of power losses respectively. These results proved that MFO algorithm gains a better result with smaller power loss than the well-known optimization methods in solving ORPD problem.

Furthermore, based on the comparison between MFO with GWO, FA and CSA for optimize results of control variables and power loss, MFO algorithm produces the lowest power loss among all these rival techniques. It is about 2.02%, 0.84% and 0.04% reduction of power loss respectively in solving ORPD problem. The comparison with other optimization techniques including flower pollination algorithm (FPA), bat algorithm (BA) and gravitational search algorithm (GSA) also have been made and presented in this paper. As a result, it is concluded that the total transmission loss obtained using MFO algorithm is outperforms as compared with other techniques.

Table 8 shows the power losses before and after optimization obtained by various methods including CSA, FA, GWO and MFO. From the table, the total transmission loss is reduced to 24.25293 MW (best case) from the base case loss of 27.864 MW by utilizing MFO algorithm. It produces 12.96% of power loss reduction which is more than the power loss reduction of GWO algorithm that produces only 11.17% from the base case loss. Moreover, it is worth to notice that all the control variables in Table 7 are within their respective ranges as stated in Table 6. The number of function evaluation (NFE) to reach the best solutions in this case study is 9000.

Fig. 7 shows the performance of MFO algorithm for various number of search agents in term of loss (MW) vs. iteration by undergoing 30 free running of simulations. From this figure, it is clearly shows that 30 search agents produce faster convergence than the other search agents. Additionally, the performance of MFO algorithm using 30 search agents for 30 free running simulations is plotted in Fig. 8 where the best, worst and average results are 24.253 MW, 26.31 MW and 24.7702 MW respectively.

Table 9
Boundary setting of control variables for IEEE 118-bus system [2,40].

Control Variables	Lower Bound	Upper Bound
Generator Bus Voltages	0.95 p.u	1.1 p.u
Load Bus Voltages	0.95 p.u	1.05 p.u
Transformer Tap Setting	0.9 p.u	1.1 p.u
Q_{C5}	−40 MVar	0 MVar
Q_{C34}	0 MVar	14 MVar
Q_{C37}	−15 MVar	0 MVar
Q_{C44}	0 MVar	10 MVar
Q_{C45}	0 MVar	10 MVar
Q_{C46}	0 MVar	10 MVar
Q_{C48}	0 MVar	10 MVar
Q_{C74}	0 MVar	12 MVar
Q_{C79}	0 MVar	20 MVar
Q_{C82}	0 MVar	20 MVar
Q_{C83}	0 MVar	10 MVar
Q_{C105}	0 MVar	20 MVar
Q_{C107}	0 MVar	6 MVar
Q_{C110}	0 MVar	6 MVar

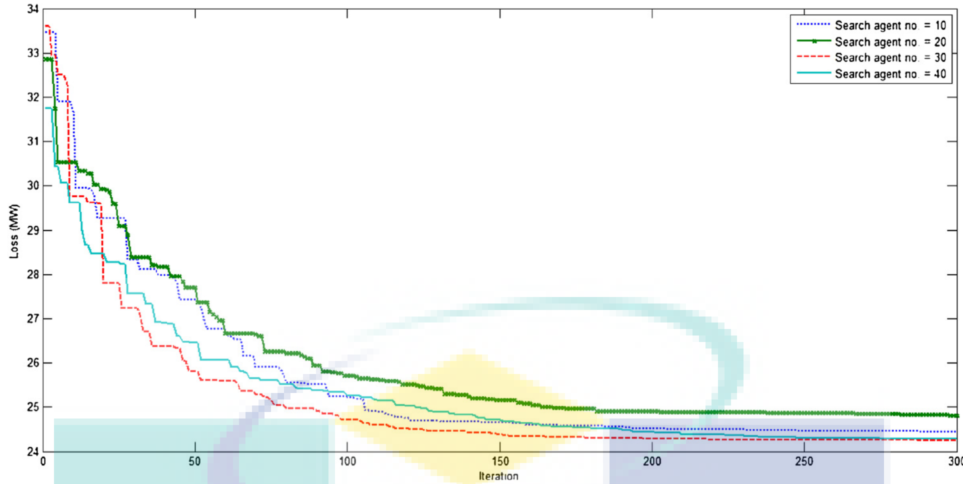

Fig. 7. Performance for different number of search agents using MFO algorithm (IEEE 57-bus system).

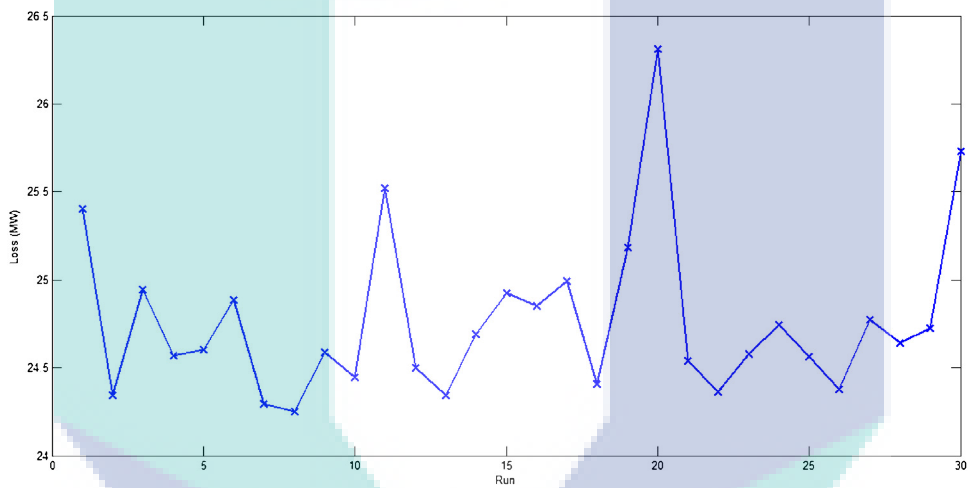

Fig. 8. Performance of 30 search agents for 30 free running simulations (IEEE 57-bus system).

5.4. IEEE 118-bus system

Undeniably, MFO algorithm has proven to be able to solve complex bus system where in this paper, IEEE 118-bus system is utilized. This system constitutes 77 control variables that need to be optimized including 54 generators, nine transformers and 14 reactive compensation elements. The minimum and maximum limits of the control variables are tabulated in Table 9. Additionally, the load demands for this case study are set as below:

$$P_{\text{Load}} = 4242\ \text{MW}$$

$$Q_{\text{Load}} = 1438\ \text{MVar}$$

The results for the best combination of control variables and the total power loss are tabulated in Table 10. By comparing the results with the best well-recognized techniques, namely PSO, DE and GA, it is obvious that MFO algorithm performs better and recorded the lowest power loss. The comparison between MFO algorithm with PSO, DE and GA gives 11.79%, 4.85% and 2.41% of loss reduction respectively. Additionally, the comparison between MFO and chaotic parallel vector evaluated interactive honey bee mating optimization (CPVEIHBMO), a new multi objective optimization algorithm [5] and GWO [2] gives 6.18% and 3.50% of loss reduction respectively. This achievement again indicated that MFO algorithm is superior than other methods.

Table 11 lists the minimum power losses and their percentage of loss reduction after optimization by using different method. For this test system, MFO algorithm is able to produce a minimum power loss of 116.4254 MW (best case) from the base case loss of 132.863 MW. It yields 12.37% of loss reduction which is more than the best result of GWO which produces only 9.19% of loss reduction from the base case loss. Furthermore, the number of function evaluation (NFE) for MFO algorithm to reach the best results in this case study is 30000. Table 12 illustrates the comparison for the simulation time in seconds of CPVEIHBMO, GSA, PSO, CLPSO, GWO and MFO.

Fig. 9 illustrates the performance of MFO algorithm with 30 search agents for 30 free running of simulations. It can be seen that the range of the losses are within 116 MW and 136 MW where the best, worst and average results are 116.44 MW, 135.96 MW and 123.9207 MW respectively. Fig. 10 shows the performance of MFO in term of loss (MW) vs. 1000 iterations for three different search agents. From the figure, it is obvious that 30 search agents get the best results among other number of search agents. In addition, 30 search agents possess the fastest convergence speed. It converge to a minimum value lower than the other number of search agents. Also, it can be noticed that as the search agent number increases, a better and faster convergence solutions will be obtained.

Table 10
Comparison of results after optimization by MFO and other selected optimization techniques for IEEE 118-bus system.

Control Variables	CPVEIHBMO [5]	GSA [41]	PSO [40]	DE [35]	GA [36]	CLPSO [40]	GWO [2]	MFO
V_1	0.9926	0.9600	1.0853	1.033570457	1.012966488	1.0332	1.0204	1.0173
V_4	1.0108	0.9620	1.0420	1.047438566	1.022099454	1.0550	1.0257	1.0402
V_6	1.0037	0.9729	1.0805	1.031598597	1.020010248	0.9754	1.0208	1.0292
V_8	0.9976	1.0570	0.9683	1.033435392	1.061467952	0.9669	1.0419	1.0600
V_{10}	1.0215	1.0885	1.0756	1.034748491	1.099378152	0.9811	1.0413	1.0374
V_{12}	1.0093	0.9630	1.0225	1.043338999	1.014803788	1.0092	1.0232	1.0250
V_{15}	1.0075	1.0127	1.0786	1.026619856	1.019613551	0.9787	1.0207	1.0268
V_{18}	1.0259	1.0069	1.0498	1.027190419	1.043221195	1.0799	1.0270	1.0298
V_{19}	0.9943	1.0003	1.0776	1.030674278	1.029458518	1.0805	1.0204	1.0275
V_{24}	1.0179	1.0105	1.0827	1.031856426	1.040458646	1.0286	1.0137	1.0483
V_{25}	1.0177	1.0102	0.9564	1.043529741	1.075946715	1.0307	1.0270	1.0600
V_{26}	0.9990	1.0401	1.0809	1.010445877	1.057209193	0.9877	1.0386	1.0600
V_{27}	1.0084	0.9809	1.0874	1.018928986	1.029248913	1.0157	1.0188	1.0267
V_{31}	0.9838	0.9500	0.9608	1.048130823	1.021541038	0.9615	1.0138	1.0101
V_{32}	0.9827	0.9552	1.1000	1.021456067	1.027054617	0.9851	1.0135	1.0226
V_{34}	1.0065	0.9910	0.9611	1.027659328	1.041382888	1.0157	1.0261	1.0556
V_{36}	1.0190	1.0091	1.0367	1.025372709	1.036107574	1.0849	1.0261	1.0548
V_{40}	1.0267	0.9505	1.0914	1.022409357	1.035039769	0.9830	1.0125	1.0419
V_{42}	0.9865	0.9500	0.9701	1.022635992	1.027421416	1.0516	1.0233	1.0429
V_{46}	1.0084	0.9814	1.0390	1.024508971	1.008306725	0.9754	1.0272	1.0450
V_{49}	1.0035	1.0444	1.0836	1.042571471	1.040791421	0.9838	1.0401	1.0589
V_{54}	0.9806	1.0379	0.9764	1.013511544	1.029785252	0.9637	1.0230	1.0284
V_{55}	0.9969	0.9907	1.0103	1.015314121	1.028853857	0.9716	1.0221	1.0289
V_{56}	0.9881	1.0333	0.9536	1.013066977	1.024740028	1.0250	1.0226	1.0283
V_{59}	1.0197	1.0099	0.9672	1.040453418	1.047253032	1.0003	1.0379	1.0512
V_{61}	0.9956	1.0925	1.0938	1.024914460	1.058332416	1.0771	1.0241	1.0534
V_{62}	1.0064	1.0393	1.0978	1.016087545	1.052064899	1.0480	1.0199	1.0506
V_{65}	0.9883	0.9998	1.0892	1.041423447	1.047670754	0.9684	1.0465	1.0596
V_{66}	1.0101	1.0355	1.0861	1.056342257	1.053088946	0.9648	1.0378	1.0600
V_{69}	0.9931	1.1000	0.9665	1.057076082	1.043874312	0.9574	1.0501	1.0600
V_{70}	1.0127	1.0992	1.0783	1.032309918	1.024484651	0.9765	1.0243	1.0600
V_{72}	1.0145	1.0014	0.9506	1.045368000	1.025167936	1.0243	1.0187	1.0526
V_{73}	1.0174	1.0111	0.9722	1.033107771	1.041474842	0.9651	1.0397	1.0600
V_{74}	1.0025	1.0476	0.9713	1.037431975	1.019569675	1.0733	1.0170	1.0600
V_{76}	0.9842	1.0211	0.9602	1.040650676	1.013382470	1.0302	1.0080	1.0390
V_{77}	0.9914	1.0187	1.0781	1.043849734	1.012598163	1.0275	1.0192	1.0502
V_{80}	1.0257	1.0462	1.0788	1.046838886	1.020977006	0.9857	1.0329	1.0600
V_{85}	0.9876	1.0491	0.9568	1.020585301	1.027345088	0.9836	1.0224	1.0600
V_{87}	1.0213	1.0426	0.9642	1.020591754	1.015471708	1.0882	1.0361	1.0599
V_{89}	1.0069	1.0955	0.9748	1.043628993	1.061188642	0.9895	1.0558	1.0600
V_{90}	1.0298	1.0417	1.0248	1.016550620	1.049040037	0.9905	1.0290	1.0431
V_{91}	0.9839	1.0032	0.9615	1.014621106	1.049951957	1.0288	1.0127	1.0496
V_{92}	1.0021	1.0927	0.9568	1.037388471	1.037043206	0.9760	1.0360	1.0600
V_{99}	0.9853	1.0433	0.9540	1.003410512	1.033165321	1.0880	1.0297	1.0551
V_{100}	1.0281	1.0786	0.9584	1.038440228	1.025139155	0.9617	1.0360	1.0584
V_{103}	0.9802	1.0266	1.0162	1.044981381	1.008421374	0.9611	1.0232	1.0442
V_{104}	1.0187	0.9808	1.0992	1.045930514	1.021911316	1.0125	1.0180	1.0333
V_{105}	1.0209	1.0163	0.9694	1.038346262	1.010179042	1.0684	1.0176	1.0281
V_{107}	1.0234	0.9987	0.9656	1.014120872	0.992917976	0.9769	1.0201	1.0161
V_{110}	0.9842	1.0218	1.0873	1.051758561	0.998231448	1.0414	1.0207	1.0215
V_{111}	1.0000	0.9852	1.0375	1.034197287	1.010311419	0.9790	1.0261	1.0280
V_{112}	0.9930	0.9500	1.0920	1.045441726	0.990803416	0.9764	1.0066	1.0042
V_{113}	1.0200	0.9764	1.0753	1.028075038	1.035600128	0.9721	1.0251	1.0350
V_{116}	1.0016	1.0372	0.9594	1.050762968	1.023853267	1.0330	1.0342	1.0484
$T_{8\text{-}5}$	1.0255	1.0659	1.0112	0.993661094	1.020567747	1.0045	1.0208	1.01360
$T_{26\text{-}25}$	0.9891	0.9534	1.0906	1.008139296	0.991029222	1.0609	1.0279	1.10000
$T_{30\text{-}17}$	0.9932	0.9328	1.0033	0.978917848	0.993920542	1.0008	1.0323	1.00380
$T_{38\text{-}37}$	0.9873	1.0884	1.0000	1.016912870	1.001403956	1.0093	1.0209	0.98263
$T_{63\text{-}59}$	0.9868	1.0579	1.0080	0.997289906	0.988192705	0.9922	1.0091	0.98430
$T_{64\text{-}61}$	1.0235	0.9493	1.0326	1.025804837	0.959802433	1.0074	1.0366	1.01390
$T_{65\text{-}66}$	1.0090	0.9975	0.9443	1.034157309	0.980371827	1.0611	1.0301	1.10000
$T_{68\text{-}69}$	1.0075	0.9887	0.9067	0.987285997	0.901094217	0.9307	1.0234	1.10000
$T_{81\text{-}80}$	0.9872	0.9801	0.9673	0.992980941	0.982876522	0.9578	1.0211	0.96831
Q_{C5}	0	0	0	−16.3153024	−19.7568825	0	−39.76	0
Q_{C34}	6.0111	7.4600	9.3639	7.942489160	8.706228539	11.7135	13.7900	0
Q_{C37}	0	0	0	−9.45279049	−8.78763477	0	−24.73	−0.03126
Q_{C44}	6.0057	6.0700	9.3078	5.875497408	9.865730545	9.8932	9.9571	10
Q_{C45}	3.0001	3.3300	8.6428	5.035965857	8.104658785	9.4169	9.8678	0
Q_{C46}	5.9838	6.5100	8.9462	3.583283672	4.364020428	2.6719	9.9186	0
Q_{C48}	3.9920	4.4700	11.8092	4.767469309	5.154589307	2.8546	14.8900	0.000842
Q_{C74}	7.9862	9.7200	4.6132	6.968724532	2.346095908	0.5471	11.9720	0.220540
Q_{C79}	13.9892	14.2500	10.5923	10.29780407	11.88268807	14.8532	19.6490	20
Q_{C82}	17.9920	17.4900	16.4544	11.66849809	10.29949342	19.4270	19.8900	0
Q_{C83}	4.0009	4.2800	9.6325	4.075634417	4.428637808	6.9824	9.9515	10
Q_{C105}	10.9825	12.0400	8.9513	5.031337342	11.56531067	9.0291	19.9680	0
Q_{C107}	2.0251	2.2600	5.0426	3.088366307	3.159801444	4.9926	5.9136	6
Q_{C110}	2.0272	2.9400	5.5319	2.694588208	2.865445523	2.2086	5.8834	6
P_{Loss} (MW)	124.098	127.76	131.99	122.3602642	119.3055517	130.96	120.65	116.4254

Table 11
Percentage of loss reduction after optimization for IEEE 118-bus system.

Compared item	Base case	CPVEIHBMO	GSA	PSO	CLPSO	DE	GA	GWO	MFO
P_{Loss} (MW)	132.863	124.098	127.76	131.99	130.96	122.3602642	119.3055517	120.65	116.4254
Percentage of loss reduction (%)	–	6.60	3.84	0.66	1.43	7.90	10.20	9.19	12.37

Table 12
Comparison of simulation time for MFO and other selected optimization algorithms on IEEE 118-bus system.

Compared item	CPVEIHBMO [5]	GSA [41]	PSO [40]	CLPSO [40]	GWO [2]	MFO
Simulation time (s)	1053.37	1198.7	1215	1472	1372	1419

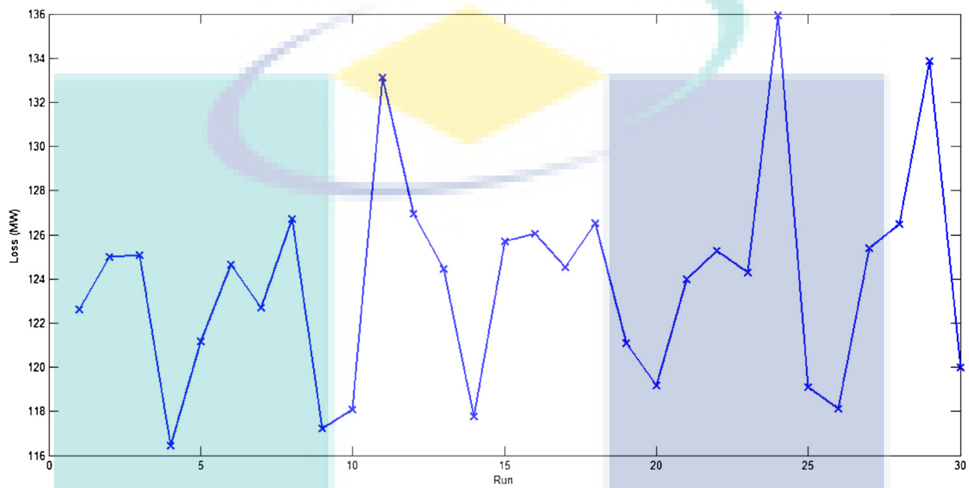

Fig. 9. Performance of 30 search agents for 30 free running simulations (IEEE 118-bus system).

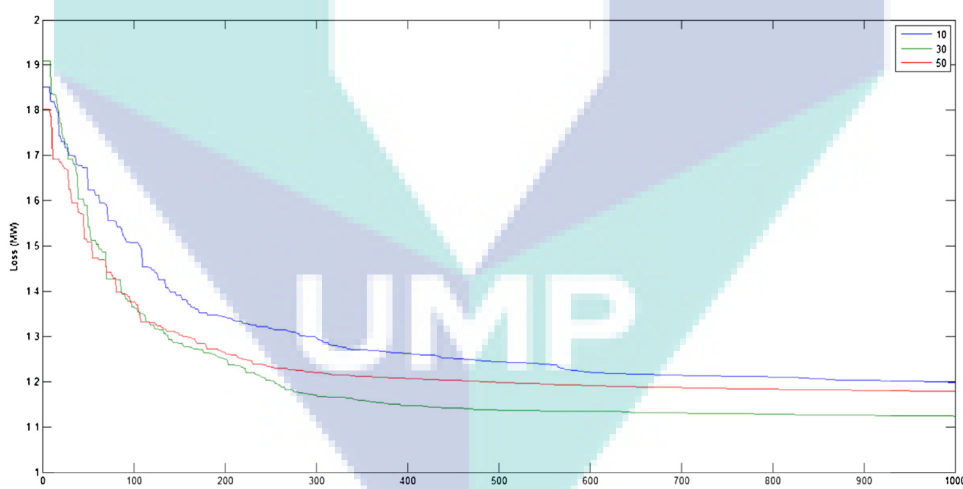

Fig. 10. Performance for different number of search agents using MFO algorithm (IEEE 118-bus system).

6. Conclusion

In this paper, the implementation of a new nature-inspired heuristic technique namely moth-flame optimizer (MFO) has been utilized to solve ORPD problem. The effectiveness and performance of this algorithm was tested using three case studies which are IEEE 30-bus system, IEEE 57-bus system and IEEE 118-bus system. The comparison between MFO with other identified methods indicates the superiority of MFO in solving the ORPD problem. It offered a novel solution with the lowest total power loss relative to other optimization methods that reported in the recent state-of-the-art literature. In this paper, it is worth to highlight that no control parameter is needed to be preset for MFO which make MFO superior in terms of simplicity. Furthermore, the minimum total transmission loss obtained using MFO will benefit the economic dispatch and secure operation of power system. In the future, MFO may be recommended as a very promising optimization technique for solving other more complex engineering problems.

Acknowledgment

This research is supported by University Malaysia Pahang (UMP) Grant #RDU150362.

References

[1] B. Zhao, C.X. Guo, Y.J. Cao, A multiagent-based particle swarm optimization approach for optimal reactive power dispatch, Power Syst. IEEE Trans. 20 (2) (2005) 1070–1078.
[2] M.H. Sulaiman, Z. Mustaffa, M.R. Mohamed, O. Aliman, Using the gray wolf optimizer for solving optimal reactive power dispatch problem, Appl. Soft Comput. 32 (2015) 286–292.
[3] A.H. Khazali, M. Kalantar, Optimal reactive power dispatch based on harmony search algorithm, Int. J. Electr. Power Energy Syst. 33 (3) (2011) 684–692.
[4] M. Ghasemi, S. Ghavidel, M.M. Ghanbarian, A. Habibi, A new hybrid algorithm for optimal reactive power dispatch problem with discrete and continuous control variables, Appl. Soft Comput. 22 (2014) 126–140.
[5] A. Ghasemi, K. Valipour, A. Tohidi, Multi objective optimal reactive power dispatch using a new multi objective strategy, Int. J. Electr. Power Energy Syst. 57 (2014) 318–334.
[6] E. Acha, H. Ambriz-Perez, C.R. Fuerte-Esquivel, Advanced transformer control modeling in an optimal power flow using Newton's method, Power Syst. IEEE Trans 15 (1) (2000) 290–298.
[7] D.S. Kirschen, H.P. Van Meeteren, MW/voltage control in a linear programming based optimal power flow, Power Syst. IEEE Trans. 3 (2) (1988) 481–489.
[8] I.P. Abril, J.A.G. Quintero, VAR compensation by sequential quadratic programming, IEEE Trans. Power Syst. 18 (1) (2003) 36–41.
[9] E.M. Soler, E.N. Asada, G.R.M. da Costa, Penalty-based nonlinear solver for optimal reactive power dispatch with discrete controls, Power Syst. IEEE Trans. 28 (3) (2013) 2174–2182.
[10] M. Varadarajan, K.S. Swarup, Differential evolution approach for optimal reactive power dispatch, Appl. Soft Comput. 8 (4) (2008) 1549–1561.
[11] P. Subbaraj, P.N. Rajnarayanan, Optimal reactive power dispatch using self-adaptive real coded genetic algorithm, Electr. Power Syst. Res. 79 (2) (2009) 374–381.
[12] N.I. Deeb, S.M. Shahidehpour, An efficient technique for reactive power dispatch using a revised linear programming approach, Electr. Power Syst. Res. 15 (2) (1988) 121–134.
[13] K. Aoki, M. Fan, A. Nishikori, Optimal VAR planning by approximation method for recursive mixed-integer linear programming, IEEE Trans. Power Syst. 3 (4) (1988) 1741–1747.
[14] K.L. Lo, S.P. Zhu, A decoupled quadratic programming approach for optimal power dispatch, Electr. Power Syst. Res. 22 (1) (1991) 47–60.
[15] S. Granville, Optimal reactive dispatch through interior point methods, IEEE Trans. Power Syst. 9 (1) (1994) 136–146.
[16] M.R. AlRashidi, M.E. El-Hawary, Applications of computational intelligence techniques for solving the revived optimal power flow problem, Electr. Power Syst. Res. 79 (4) (2009) 694–702.
[17] S. Durairaj, D. Devaraj, P.S. Kannan, Genetic algorithm applications to optimal reactive power dispatch with voltage stability enhancement, J. Inst. Eng. India Part El Electr. Eng. Div. 87 (42) (2006).
[18] Q.H. Wu, Y.J. Cao, J.Y. Wen, Optimal reactive power dispatch using an adaptive genetic algorithm, Int. J. Electr. Power Energy Syst. 20 (8) (1998) 563–569.
[19] Q.H. Wu, J.T. Ma, Power system optimal reactive power dispatch using evolutionary programming, IEEE Trans. Power Syst. 10 (3) (1995) 1243–1249.
[20] K. Ayan, U. Kılıç, Artificial bee colony algorithm solution for optimal reactive power flow, Appl. Soft Comput. 12 (5) (2012) 1477–1482.
[21] J.G. Vlachogiannis, K.Y. Lee, A comparative study on particle swarm optimization for optimal steady-state performance of power systems, Power Syst. IEEE Trans. 21 (4) (2006) 1718–1728.
[22] N. Sinsuphan, U. Leeton, T. Kulworawanichpong, Optimal power flow solution using improved harmony search method, Appl. Soft Comput. 13 (5) (2013) 2364–2374.
[23] K. Valipour, A. Ghasemi, Using a new modified harmony search algorithm to solve multi-objective reactive power dispatch in deterministic and stochastic models, J. AI Data Min. 5 (1) (2017) 89–100.
[24] S. Duman, U. Güvenç, Y. Sönmez, N. Yörükeren, Optimal power flow using gravitational search algorithm, Energy Convers. Manage. 59 (2012) 86–95.
[25] B. Shaw, V. Mukherjee, S.P. Ghoshal, Solution of reactive power dispatch of power systems by an opposition-based gravitational search algorithm, Int. J. Electr. Power Energy Syst. 55 (2014) 29–40.
[26] C. Dai, W. Chen, Y. Zhu, X. Zhang, Reactive power dispatch considering voltage stability with seeker optimization algorithm, Electr. Power Syst. Res. 79 (10) (2009) 1462–1471.
[27] C.-M. Huang, Y.-C. Huang, Combined differential evolution algorithm and ant system for optimal reactive power dispatch, Energy Procedia 14 (2012) 1238–1243.
[28] M. Mehdinejad, B. Mohammadi-Ivatloo, R. Dadashzadeh-Bonab, 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. 83 (2016) 104–116.
[29] M. Basu, Quasi-oppositional differential evolution for optimal reactive power dispatch, Int. J. Electr. Power Energy Syst. 78 (2016) 29–40.
[30] S.M. Mohseni-Bonab, A. Rabiee, B. Mohammadi-Ivatloo, S. Jalilzadeh, S. Nojavan, A two-point estimate method for uncertainty modeling in multi-objective optimal reactive power dispatch problem, Int. J. Electr. Power Energy Syst. 75 (2016) 194–204.
[31] D.H. Wolpert, W.G. Macready, No free lunch theorems for optimization, Evol. Comput. IEEE Trans. 1 (1) (1997) 67–82.
[32] S. Mirjalili, Moth-flame optimization algorithm: a novel nature-inspired heuristic paradigm, Knowl.-Based Syst. 89 (2015) 228–249.
[33] R.D. Zimmerman, C.E. Murillo-Sanchez, R.J. Thomas, MATPOWER: steady-state operations, planning, and analysis tools for power systems research and education, Power Syst. IEEE Trans. 26 (1) (2011) 12–19.
[34] K.D. Frank, C. Rich, T. Longcore, Effects of artificial night lighting on moths, Ecol. Consequences Artif. Night Light (2006) 305–344.
[35] K. Price, R.M. Storn, J.A. Lampinen, Differential Evolution: A Practical Approach to Global Optimization, Springer Science & Business Media, 2006.
[36] X. Blasco, Control predictivo basado en modelos mediante técnicas de optimización heurística. Aplicación a procesos no lineale sy multivariables. Tesis Doctoral, Universidad Politécnica de Valencia, España, 1999.
[37] A.A.A. El Ela, M.A. Abido, S.R. Spea, Differential evolution algorithm for optimal reactive power dispatch, Electr. Power Syst. Res. 81 (2) (2011) 458–464.
[38] M.H. Sulaiman, M.I. Mohd Rashid, O. Aliman, M.R. Mohamed, A.Z. Ahmad, M.S. Bakar, Loss minimisation by optimal reactive power dispatch using cuckoo search algorithm, 3rd IET International Conference on Clean Energy and Technology (CEAT) 2014 (2014) 1–4.
[39] M.H. Sulaiman, Z. Mustaffa, H. Daniyal, M.R. Mohamed, O. Aliman, Solving optimal reactive power planning problem utilizing nature inspired computing techniques:ARPN, J. Eng. Appl. Sci. 10 (2015) 9779–9785.
[40] K. Mahadevan, P.S. Kannan, Comprehensive learning particle swarm optimization for reactive power dispatch, Appl. Soft Comput. 10 (2) (2010) 641–652.
[41] S. Duman, Y. Sonmez, U. Guvenc, N. Yorukeren, Optimal reactive power dispatch using a gravitational search algorithm, IET Gener. Transm. Distrib. 6 (6) (2012) 563–576.

An Application of Multi-Verse Optimizer for Optimal Reactive Power Dispatch Problems

Mohd Herwan Sulaiman
Fakulti Kejuruteraan Elektrik & Elektronik
Universiti Malaysia Pahang
26600 Pekan, Pahang
mherwan@ieee.org

Zuriani Mustaffa
Fakulti Kejuruteraan Sistem Komputer & Kejuruteraan Perisian
Universiti Malaysia Pahang
26300 Gambang, Pahang
zuriani@ump.edu.my

Mohd Rusllim Mohamed
Fakulti Kejuruteraan Elektrik & Elektronik
Universiti Malaysia Pahang
26600 Pekan, Pahang
rusllim@ump.edu.my

Omar Aliman
Fakulti Kejuruteraan Elektrik & Elektronik
Universiti Malaysia Pahang
26600 Pekan, Pahang
omaraliman@ieee.org

***Abstract* — This paper proposes a new algorithm namely Multi-Verse Optimizer (MVO) in solving the Optimal Reactive Power Dispatch (ORPD) problem. It is inspired from the three main concepts in cosmology viz. white hole, black hole and wormhole. These concepts are developed mathematically to perform exploration, exploitation and local search respectively. This algorithm is applied to obtain the best combination of control variables such as generator voltages, tap changing transformer's ratios, reactive compensation devices as well as real power generation. In this paper, to show the effectiveness of MVO into ORPD problem, IEEE-30 bus system with 25 control variables is utilized and compared with recent algorithms available in literature. The result of this study shows that MVO is able to achieve less power loss than those determined by other techniques.**

Keywords - loss minimization; multi-verse optimizer; nature inspired algorithms; optimal reactive power dispatch

I. Introduction

Optimal reactive power dispatch (ORPD) is one of the nonlinear and non-convex problems in power system planning and operation. Control variables or parameters for ORPD normally have close relationship with reactive power flow such as voltage magnitudes of generator buses, transformer tap ratios and reactive compensation elements [1]. In literature, there are several objective functions that have been addressed and assessed to achieve the successful of ORPD such as loss, voltage deviation and voltage stability index minimizations [2]. Nevertheless, for this paper, only loss minimization is used for objective function to overcome the ORPD problem. In order to achieve this objective, the stated control variables need to be controlled and set accordingly.

It is a nonlinear problems and difficult task since all the controlled variables need to be set simultaneously to achieve the minimum loss. That is why there are massive researches have been done to overcome this problem such as by using classical techniques including Newton techniques [3], sequential quadratic programming [4] and non-linear solver with penalty based [5].

Recently, many nature inspired algorithms have been proposed to solve ORPD such as grey wolf optimizer (GWO) [6], artificial bee colony (ABC) [1], harmony search algorithm (HSA) [2], particle swarm optimization (PSO) [7], honey bee mating optimization (HBMO) [8], gravitational search algorithm (GSA) [9] and many more to come.

This paper proposes the recent algorithm based on the universe cosmology concepts to solve ORPD problem. This algorithm has been proposed by [10]. The organization of this paper is as follows: Section 2 presents the ORPD formulation while brief description of MVO is discussed in Section 3. It is followed by the implementation of MVO into solving ORPD problem in Section 4. Section 5 presents the results and discussion and finally the conclusion is stated in Section 6.

II. Optimal Reactive Power Dispatch

ORPD problem is one of the most complex problems in power engineering system which can be described as the minimization of function f(x, u) subject to the following expressions:

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

where $g(x,u)$ and $h(x,u)$ are the equality and inequality constraints respectively, x is the dependent variables and u is the control variables. In this paper, the objective function of $f(x, u)$ is to minimize the transmission loss system.

The equality constraint equation is the power balanced of load flows which are expressed as follow [2]:

$$
\begin{aligned}
P_{Gi} - P_{Di} &= V_i \sum V_j \left(G_{ij} \cos\theta_{ij} + B_{ij} \sin\theta_{ij} \right) \\
Q_{Gi} - Q_{Di} &= V_i \sum V_j \left(G_{ij} \cos\theta_{ij} - B_{ij} \sin\theta_{ij} \right)
\end{aligned} \tag{2}
$$

The inequality constraints are represented in terms of operating constraints such as generators' constraints (upper and lower bound), transformer tap setting as well as reactive elements' upper and lower limits, expressed as follow [6]:

$$P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max} \qquad i = 1,\ldots,N_G \tag{3}$$

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

$$T_i^{\min} \le T_i \le T_i^{\max} \qquad i = 1,\ldots,N_T \tag{5}$$

$$Q_{Ci}^{\min} \le V_{Ci} \le V_{Ci}^{\max} \qquad i = 1,\ldots,N_C \tag{6}$$

where N_G, N_T and N_C are number of generators, number of transformers and number of shunt compensators respectively. It is worth to highlight that in this paper that the MATPOWER software package [11] is utilized to obtain total transmission loss by running the load flow program in order to obtain the precise result.

III. MULTI-VERSE OPTIMIZER

MVO algorithm is inspired by the concept of multi-verse theory which consists of three verses: white holes, black holes and wormholes. In this algorithm, a population based is divided the search process into two phases: exploration and exploitation. White hole and black hole concepts are used as exploration and wormhole is treated as exploitation in this algorithm.

There are rules have been applied in MVO to the universe [10]:

a) The higher inflation rate, the higher probability of having white hole.

b) The higher inflation rate, the lower probability of having black holes.

c) Universes with higher inflation rate tend to send objects through white holes.

d) Universes with lower inflation rate tend to receive more objects through black holes.

e) The objects in all universes may face random movement towards the best universe via wormholes regardless of the inflation rate.

The conceptual of this algorithm is depicted in Fig 1.

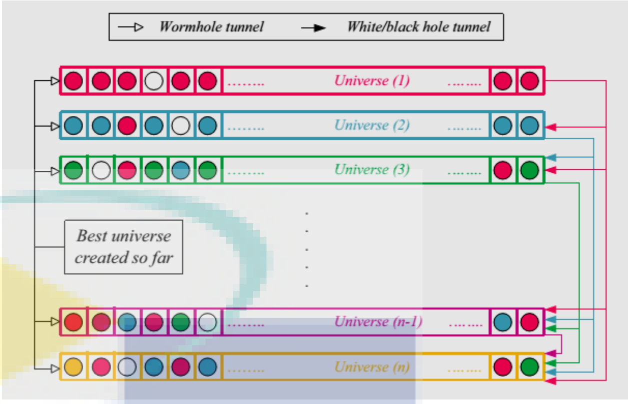

Figure 1. Conceptual of MVO algorithm [10].

The development of MVO initially can be described as the following expression:

$$
U = \begin{bmatrix}
x_1^1 & x_1^2 & \cdots & x_1^d \\
\vdots & \vdots & \cdots & \vdots \\
\vdots & \vdots & \cdots & \vdots \\
x_n^1 & x_n^2 & \cdots & x_n^d
\end{bmatrix} \tag{7}
$$

where U is a set of solution, d is the number of variables (dimension) and n is the number of universes (candidate of solution) which can be described as follows:

$$
x_i^j = \begin{cases} x_k^j & r1 < NI(U_i) \\ x_i^j & r1 \ge NI(U_i) \end{cases} \tag{8}
$$

where x_i^j is the jth parameter of ith universe, Ui shows the ith universe, $NI(U_i)$ is normalized inflation rate of the ith universe, $r1$ is a random number between 0 and 1, and x_k^j is the jth parameter of kth universe selected by a roulette wheel selection mechanism.

The universes keep exchanging objects without perturbation. To maintain the diversity of universes and to exploit the searching process in MVO, each universe is treated to have wormholes to transport its objects through space randomly. In order to provide local changes for each universe and have high probability of improving the inflation rate using wormholes, that particular wormhole tunnels are always established between a universe and the best universe formed so far, which can be described as follows:

$$
x_i^j = \begin{cases} \begin{cases} X_j + TDR \times \left(\left(ub_j - lb_j \right) \times + lb_j \right) & r3 < 0.5 \\ X_j - TDR \times \left(\left(ub_j - lb_j \right) \times + lb_j \right) & r3 \ge 0.5 \end{cases} & r2 < WEP \\ x_i^j & r2 \ge WEP \end{cases} \tag{9}
$$

where X_j is the jth parameter of best universe formed so far, *TDR* (travelling distance rate) and *WEP* (wormhole existence probability) are coefficients, lb_j and ub_j are the lower bound and upper bound of jth variable respectively, x_i^j is the jth parameter of ith universe and $r2$, $r3$, $r4$ are random numbers

DOI 10.5013/IJSSST.a.17.41.05

ISSN: 1473-804x online, 1473-8031 print

between 0 and 1. *WEP* and *TDR* are treated as adaptive formula as follow:

$$WEP = \min + l \times \left(\frac{\max - \min}{L} \right) \tag{10}$$

$$TDR = 1 - \frac{l^{1/p}}{L^{1/p}} \tag{11}$$

where *min* is the minimum (for this paper is set to 0.2), *max* is the maximum (for this paper is set to 1), l is the current iteration, L is the maximum iterations and p is the exploitation accuracy over the iterations (for this paper is set to 6). The higher p, the sooner and more accurate exploitation/ local search [10].

It is worth to highlight that the MVO algorithm depends on number of iterations, number if universes, roulette wheel mechanism and universe sorting mechanism. Quicksort algorithm is used to sort universe at each iteration and roulette wheel selection is run for every variable in every universe over iterations. Details description of MVO can be obtained in [10].

IV. MVO FOR ORPD PROBLEM

The application of MVO in solving the ORPD is to find the optimal combination of control variables in order to achieve loss minimization by fulfilling all the constraints mentioned in section 2. Initially, the number of universe or search agents and maximum iteration are set. The universes (candidate for solution) is constructed in matrix form as depicted in eqn. (7) where the row represents the number of universes and the column represents the number of control variables to be optimized.

To obtain the objective function, each universe is mapped into the load flow data and then the load flow program using MATPOWER software package is executed to find the total transmission loss. Once the loss has obtained for respected universe (after updating the variables using eqns. (8-9)), the matrix is sorted where the best solution so far is located at the top while the worst result is located at the bottom of the population matrix. The updated position process is done by using roulette wheel selection. If the updated variables are out of bound from the constraints, they are pegged at the minimum or maximum boundaries so that the result obtained is correct. The implementation of MVO in solving ORPD is depicted in Fig. 2.

V. RESULTS AND DISCUSSION

In order to show the veracity and effectiveness of MVO in solving ORPD problems, the IEEE-30 bus system is utilized as the test system in this paper. The simulation was implemented in MATLAB.

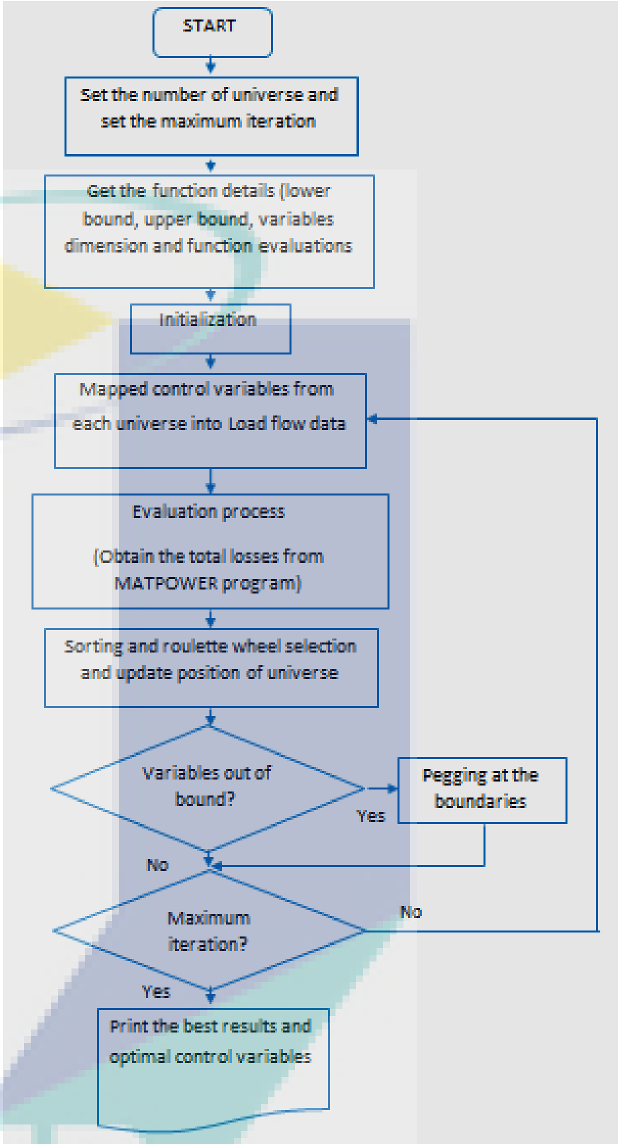

Figure 2. Flow of proposed MVO for solving ORPD problem

The IEEE 30-bus system is based on [1] which consists of 25 control variables that need to be optimized. This system consists of six generators, 41 lines, four transformers that located at lines 6-9, 4-12, 9-12 and 27-28 and three shunt reactive elements located at buses 10, 12, 5, 17, 20, 21, 23, 24 and 29. For this study, the additional control variables need to be optimized which is the generation production by the generator buses. The real and reactive load demand for this study is set to 283.2 MW and 126.2 MVar respectively.

For fair comparison, the results presented in [1, 6] are mapped into the similar load flow program to investigate the total transmission losses. Table 1 shows the results obtained using ABC [1], GWO [6] as well as MVO algorithm. It can be noted that the total loss obtained by MVO is the best compared to ABC and GWO. It is about 3.6% reduction of

ISSN: 1473-804x online, 1473-8031 print

total loss if MVO is compared with the ABC and MVO is slightly better compared to GWO. This table also shows that all control variables for all algorithms converged within their respective limits.

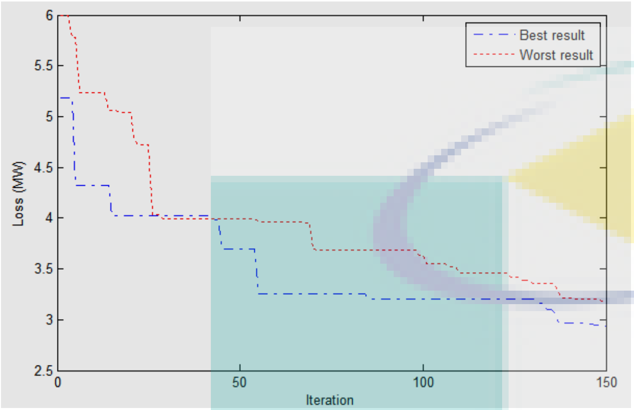

Figure 3. Performance for the best and worst results of ORPD using MVO.

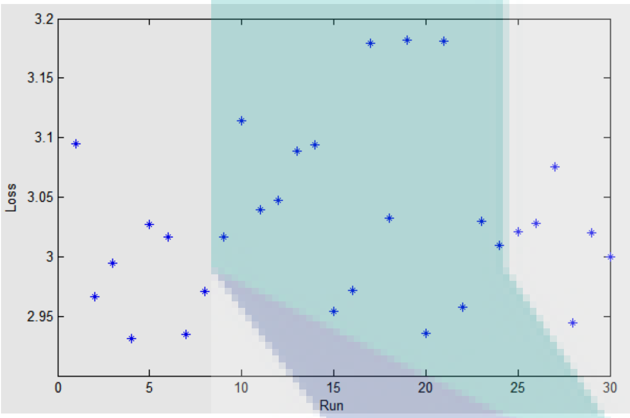

Figure 4. Performance for 30 universes for 30 free running simulations.

To exhibit the effectiveness of the proposed MVO, the performance of this algorithm in terms of best and worst results in terms of loss (MW) versus iterations is shown in Fig. 3. The performance of GWO for 30 running simulations is depicted in Fig. 4. The results of total loss are varied between 2.9 MW and 3.2 MW. The best and the worst results are recorded at 2.9311 MW and 3.1818 MW respectively. Fig. 5 shows the performance for various numbers of agent (universes) of MVO. It is worth to highlight that the results presented in this figure is only the best result of each number of universes among 30 free running simulations. It can be concluded that 30 universes is adequate to obtain good combination of control variables of ORPD within 150 iterations.

TABLE I. RESULTS FOR OPTIMIZED CONTROL VARIABLES FOR IEEE 30-BUS SYSTEM

Control Variables	Limits		ABC [1]	GWO [6]	MVO
	Lower	Upper			
P_1	0.5	2	0.5462	0.516117	0.5266
P_2	0.2	0.8	0.7863	0.79793	0.79902
P_5	0.15	0.5	0.4903	0.5	0.48827
P_8	0.1	0.35	0.3477	0.34933	0.34968
P_{11}	0.1	0.3	0.2999	0.3	0.30
P_{13}	0.12	0.4	0.3945	0.4	0.39974
V_1	1	1.1	1.0927	1.1	1.1
V_2	1	1.1	1.088	1.0981	1.0988
V_5	1	1.1	1.0695	1.0766	1.0833
V_8	1	1.1	1.0722	1.087	1.089
V_{11}	1	1.1	1.086	1.097	1.0999
V_{13}	1	1.1	1.0926	1.1	1.1
T_1	0.9	1.1	0.9983	0.9912	1.0446
T_2	0.9	1.1	0.9994	1.0402	0.95684
T_3	0.9	1.1	0.9984	1.0332	0.9889
T_4	0.9	1.1	1.0034	0.99125	0.98658
QC_{10}	0	0.05	0.0155	0.043587	0.04739
QC_{12}	0	0.05	0.0394	0.010303	0.030453
QC_{15}	0	0.05	0.0347	0.026824	0.027326
QC_{17}	0	0.05	0.0331	0.05	0.036541
QC_{20}	0	0.05	0.0332	0.00058404	0.045029
QC_{21}	0	0.05	0.0395	0.030001	0.04825
QC_{23}	0	0.05	0.013	0.0056929	0.041027
QC_{24}	0	0.05	0.0371	0.045864	0.039961
QC_{29}	0	0.05	0.0399	0.0043827	0.047437
Total Loss (MW)			**3.041**	**2.9377**	**2.9311**

VI. CONCLUSION

This paper has proposed a recent nature inspired computing algorithm, Multi-Verse Optimizer algorithm in solving ORPD problem. The effectiveness of MVO was demonstrated using IEEE 30-bus system. Simulation results showed that MVO is better compared to other selected algorithms in terms of finding the minimum power loss. The implementation of MVO into other objective functions such as voltage deviation as well as including the practical constraints related to generating units will be proposed in the near future.

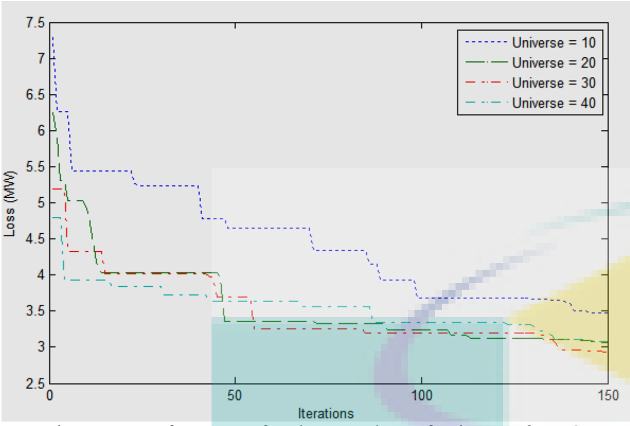

Figure 5. Performance of various numbers of universe of MVO.

ACKNOWLEDGMENT

This research study is supported by Universiti Malaysia Pahang under UMP Research Grant #150362.

REFERENCES

[1] 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, May 2012

[2] 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, Mar. 2011.

[3] E. Acha, H. Ambriz-Perez and C. R. Fuerte-Esquivel, "Advanced Transformer Control Modeling in an Optimal Power Flow Using Newton's Method," IEEE Transactions on power Systems, vol. 15, no. 1 February 2000, pp. 290-298.

[4] I. P. Abril and J. A. G. Quintero, "VAR compensation by sequencial quadratic programming," IEEE Transactions on Power Systems, vol. 18, Dec. 2003, pp. 36-41.

[5] E. M. Soler, E. N. Asada and R. M. da Costa, "Penalty-based nonlinear solver for optimal reactive power dispatch with discrete controls," IEEE Transacations on Power Systems, vol. 28, no. 3. August 2013, pp. 2174-2182.

[6] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, "Using the gray wolf optimizer for solving optimal reactive power dispatch problem," Applied Soft Computing, vol. 32, no. 0, pp. 286–292, 2015.

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

[8] 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, Sep. 2014.

[9] 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, Feb. 2014

[10] S. Mirjalili, S. M. Mirjalili, "Multi-Verse Optimizer: a nature-inspired algorithm for global optimization," Neural Computing & Applications, pp 1-19, March 2015.

[11] Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J., "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," IEEE Transactions on Power Systems, vol.26, no.1, pp.12-19, Feb. 2011.

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 setting 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 meta-heuristic algorithms. The results of this paper illustrate that MFO is able to achieve a lower power loss than ALO and other selected algorithms from literatures.**

***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 problems 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 meta-heuristic 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 to 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:

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

where:

$f(x,u)$	= Objective function
$g(x,u)=0$	= Equality constraints
$h(x,u)\le 0$	= 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):

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

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

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
θ_{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:

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

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

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

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

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

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:

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

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

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]:

$$M_i = S(M_i, F_j) \quad (11)$$

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:

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

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:

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

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 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 continued until the stopping criterion (maximum iteration) is reached. The application of MFO and ALO in solving ORPD is illustrated in Figure-1.

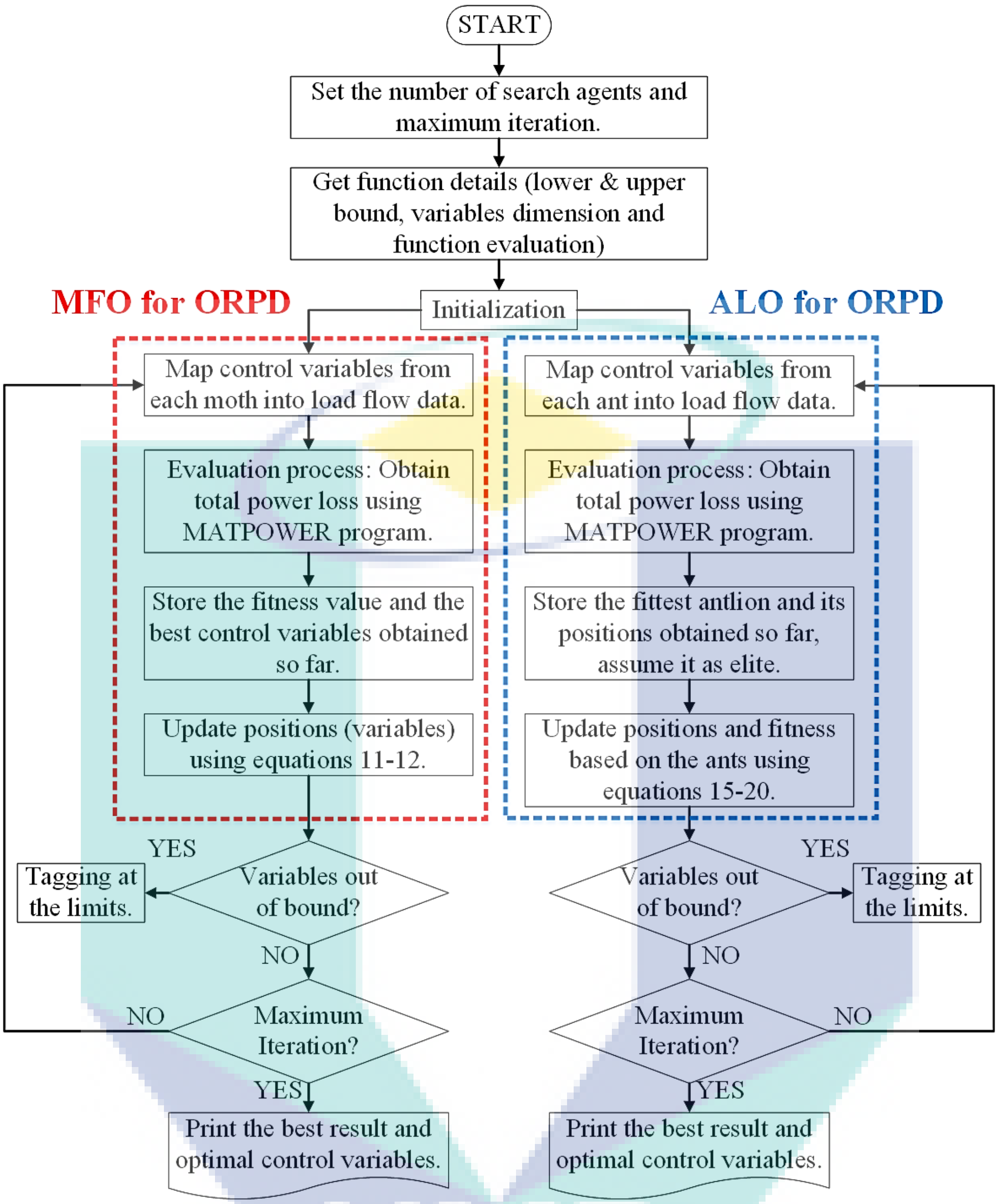

Figure 1: Flowchart of MFO and ALO for solving ORPD

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 are 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.

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 a 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 produces a lower range of power losses than ALO. However, ALO can produces a 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

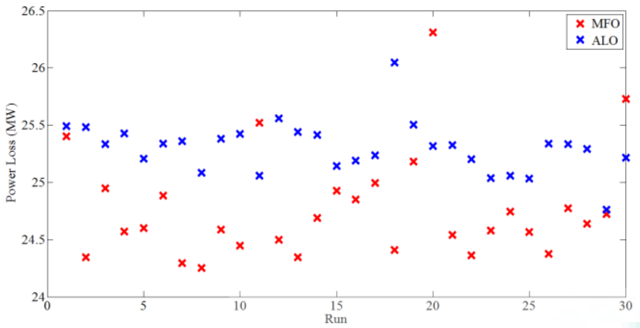

Figure 2: Comparison of power loss performances between ALO and MFO for 30 trail runs

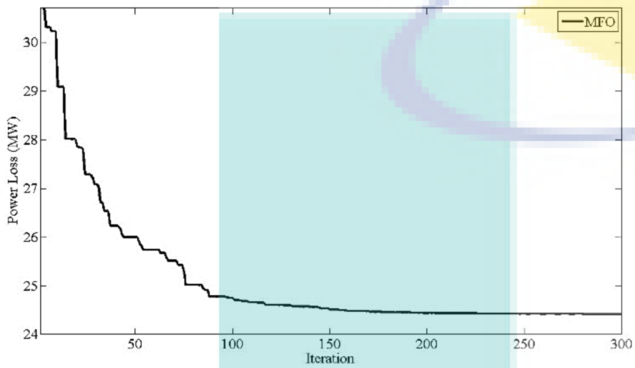

Figure 3: Convergence performance of MFO for power loss minimization

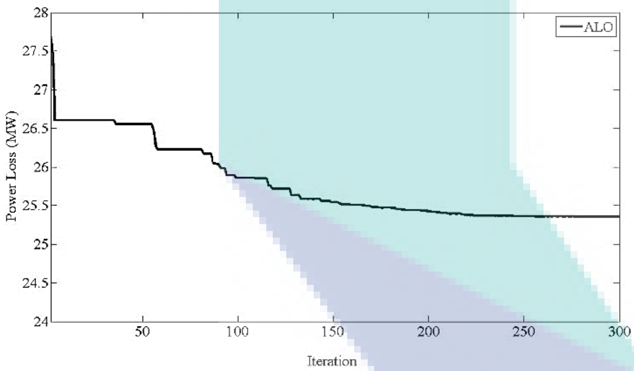

Figure 4: Convergence performance of ALO for power loss minimization

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 were 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 literatures in terms of obtaining the lowest power loss. Whereas, ALO is the worst among the compared algorithms. However, ALO can produces a 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.

Cuckoo Search Algorithm as an Optimizer for Optimal Reactive Power Dispatch Problems

Mohd Herwan Sulaiman
Fakulti Kejuruteraan Elektrik & Elektronik
Universiti Malaysia Pahang
Pekan, Pahang, Malaysia
e-mail: mherwan@ieee.org

Zuriani Mustaffa
Fakulti Sistem Komputer & Kejuruteraan Perisian
Universiti Malaysia Pahang
Gambang, Pahang, Malaysia
e-mail: zuriani@ump.edu.my

***Abstract*—This paper presents the application of Cuckoo Search Algorithm (CSA) in optimizing the control variables of power system operation in solving the optimal reactive power dispatch (ORPD) problem. CSA is inspired by the parasitic behavior of Cuckoo birds in reproduction process based on the probability for a host bird in discovering an alien egg in its nest. The implementation of CSA in determining the optimal value of control variables such as generator bus voltages, transformer tap setting and shunt reactive elements in order to obtain the minimize loss in the system. In this paper, IEEE-30 bus system is utilized to show the effectiveness of CSA and then the comparison with other nature inspired algorithms will be presented.**

Keywords-cuckoo search algorithm; loss minimization; nature inspired algorithms; optimal reactive power dispatch

I. INTRODUCTION

Optimal Reactive Power Dispatch (ORPD) is one of the complex problems in power system planning and operation which can be treated as a sub-problem of optimal power flow (OPF) problems. One of the main objectives of ORPD is to obtain a minimum power loss in the system by configuring the control variables while fulfilling all the system's constraints. The control variables that need to be optimized are voltage magnitude of generator buses, transformer tap setting as well as shunt reactive elements.

Recently, there are various nature inspired algorithms that have been applied to solve the ORPD problems such as Artificial Bee Colony (ABC), Grey Wolf Optimizer (GWO) [2], Gravitational Search Algorithm (GSA) [3], Chaotic Krill Herd Algorithm (CKHA) [4] and many more.

This paper intends to propose a nature inspired based algorithm namely Cuckoo Search Algorithm (CSA) in order to obtain an optimal solution of ORPD problem especially in loss minimization problem. Even though there is a work has been done in similar field such as discussed in [5], the different approach has been taken in this study which is the integration between CSA with MATPOWER software package [6] so that the accurate and precise loss calculation can be obtained. CSA on the other hand is a nature inspired algorithm proposed by Yang and Deb [7] in 2009 that mimic the brood parasitism behavior of Cuckoo birds.

The application of CSA into other optimization problems has been done such as in combined heat and economic dispatch [8], advanced machining process [9], multi-pass turning operations [10], selection of obsolete tools in manufacturing process [11] as well as in optimal replacement policy for obsolete components [12].

This paper is organized as follows: in Section 2, the formulation of ORPD for loss minimization is briefly discussed followed by the CSA in general is presented in Section 3. CSA implementation for solving ORPD is discussed in Section 4. Section 5 presents the results and discussion and finally, the conclusion is stated in Section 6.

II. OPTIMAL REACTIVE POWER DISPATCH FOR LOSS MINIMIZATION

One of the objectives of ORPD is the loss minimization which can be represented as follows:

$$\min f(x,u) = P_{Loss} \tag{1}$$

where x is the vector of dependent variables and u is the vector of control variables. The function f is subject to the equality and inequality constraints. The equality constraint is the power balanced equation which is can be expressed as follow:

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

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

where P_{Gi} and Q_{Gi} are the real and reactive power generation at bus i, P_{Di} and Q_{Di} are the real and reactive load at bus i. V_i and V_j are the voltage magnitude of ith and jth bus, G_{ij} and B_{ij} are the conductance and susceptance of transmission line i-j, and θ_{ij} is the angle difference of i-jth transmission line.

The inequality constraints can be listed down as follow:

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

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

$$T_i^{\min} \le T_i \le T_i^{\max} \qquad i = 1,\ldots,N_T \tag{6}$$

$$Q_{Ci}^{\min} \le Q_{Ci} \le Q_{Ci}^{\max} \qquad i = 1,\ldots,N_C \tag{7}$$

where equations (4) and (5) represent the generator constraints in terms of generators' output and voltages respectively, equation (6) is for the transformer tap setting boundaries and equation (7) is for the injected shunt reactive element's boundaries. N_G, N_T and N_C are number of generators, number of transformers and number of shunt compensators respectively. In this paper, the MATPOWER software package [6] is utilized to obtain total transmission loss by running the load flow program in order to obtain the precise result.

III. Cuckoo Search Algorithm

Cuckoo Search Algorithm (CSA) is a nature inspired algorithm developed by Yang and Deb in 2009 [7] which is based on the brood parasitism of cuckoo birds. Cuckoo birds have a unique aggressive reproduction strategy. Based on this strategy, Yang and Deb have come out with the three idealized rules:

a) Each cuckoo lays one egg at a time and leave it egg in a randomly chosen nest.

b) The best nest with high quality eggs will be carried over to the next generations.

c) The number of available host nests is fixed and the egg laid by a cuckoo is discovered by the host bird with a probability ρa which is between 0 and 1. The host bird can either get rid of the egg or simply leave the nest to build a new nest.

In addition, the integration of Levy flight has been done in this algorithm to enhance the searching capability. It can be expressed as follows:

$$x_i^{(t+1)} = x_i^t + \alpha \oplus Levy(\lambda) \tag{8}$$

where $\alpha > 0$ is the step size that normally lies within the range of the problem's search space. The Levy flight provides a random walk which is derived from the Levy distribution, as follows:

$$Levy \sim u = t^{-\lambda}, \qquad (1 < \lambda \leq 3) \tag{9}$$

IV. CSA Implementation for ORPD

The proposed CSA for solving ORPD problem especially in finding the optimal solution of control variables so that the loss minimization can be achieved with all constraints are satisfied. The implementation of CSA into ORPD solution is done as follows:

Step 1: The discovery rate of alien eggs, ρ_a and the maximum iteration are set. Then the initialization of the candidate of solution is set randomly. Let

$$X = \begin{bmatrix} x_1^1 & \cdots & P_n^1 \\ \vdots & \ddots & \vdots \\ X_1^p & \cdots & P_n^p \end{bmatrix} \tag{10}$$

where p is the number of population or search agents and n is the dimension of the control variables to be optimized. Each element in the population should satisfy the constraints in (4) - (7).

Step 2: Evaluation process. Each population that contains initial value of control variables (voltage magnitude for generators, transformer tap setting and reactive elements) is mapped into the load flow data in MATPOWER and the load flow program is run to calculate the power flow solution including the power losses. Each population will undergo the same process in order to record the power loss in the system. At this point, the best result so far is recorded by the CSA program.

Step 3: Generation of new solutions by using equations (8-9) but keep the current best. Check boundaries of the variables. If out of bound, the program will choose pegged at the boundary values.

Step 4: Evaluation process for the new generated solutions. Record the best objective (minimum loss) so far.

Step 5: Discovery process. Replace some nests by constructing new solutions by referring to the rate of alien eggs, ρ_a.

Step 6: Evaluation process for new set of solutions.

Step 7: Find the best results so far and record the solutions.

Step 8: Repeat Steps 3-7 until the stopping criterion is reached (maximum iteration).

Step 9: Print the optimal results and the best objective values. General implementation of CSA into ORPD is depicted in Fig. 1.

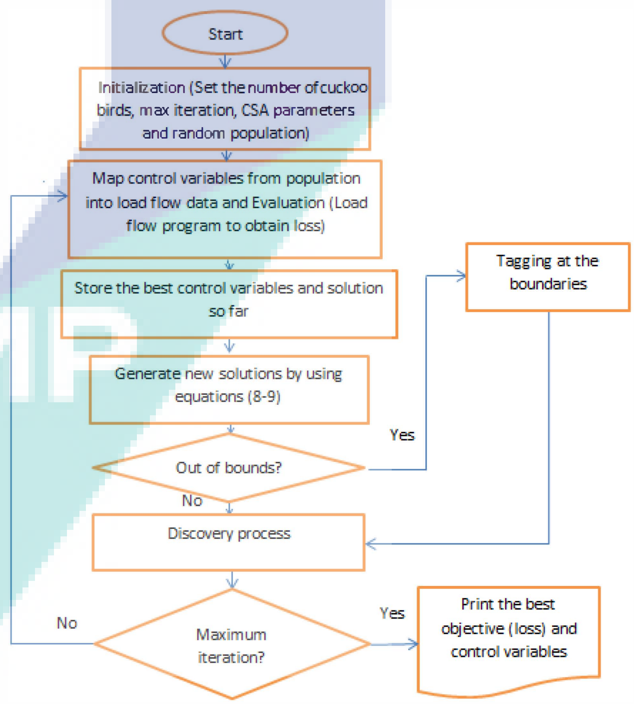

Figure 1. Implementation of CSA in solving the ORPD problem.

V. Results and Discussion

In order to show the effectiveness and veracity of the proposed CSA, IEEE-30 bus system is used where this system consists of six generators, 41 transmission lines, four transformers and three injected shunt reactive elements located at buses 3, 10 and 24. For this study, the real and reactive demands are 283.4 MW and 126.20 MVAR respectively.

The performance of CSA then is compared with five other recent nature inspired algorithms viz. Moth Flame Optimizer (MFO) [13], Grey Wolf Optimizer (GWO) [2], Firefly Algorithm (FA) [14], Flower Pollination Algorithm (FPA) [15] and Gravitational Search Algorithm (GSA) [3]. All these algorithms are programmed and run in MATLAB so that the performance of each algorithm can be analyzed and compared fairly. For fair comparison, all the parameters are set similarly for all algorithms, where for this case, the number of search agents is set to 30 and the number of maximum iteration is set to 150.

Table I shows the boundaries setting of control variables for this system. It can be seen that the generator's voltage magnitude is slightly different compared to the one that has been presented in [2] where the limit is set to 0.95 for lower bound and 1.05 for upper bound. This is due to the 5 % tolerance is set for the voltage magnitude instead of 10% tolerance which has been applied in [2].

Table II shows the detail results obtained for CSA, MFO, GWO, FA, FPA and GSA. From this table, it can be seen that CSA, MFO and GWO obtained the minimum loss, 5.0667 MW compared to FA, FPA and GSA. It can be conclude that FPA gave the worst results among all algorithms tested in this system. It also can be noted that all algorithms able to obtain the variables within the specified boundaries shown in Table I.

TABLE I. BOUNDARIES SETTING OF CONTROL VARIABLES FOR IEEE-30 BUS SYSTEM

Control Variables	Lower Bound	Upper Bound
Generator's Voltage Magnitude	0.95 p.u	1.05 p.u
Transformer Tap Setting	0.95 p.u	1.05 p.u
Reactive Shunt Elements	-12 MVar	36 MVar

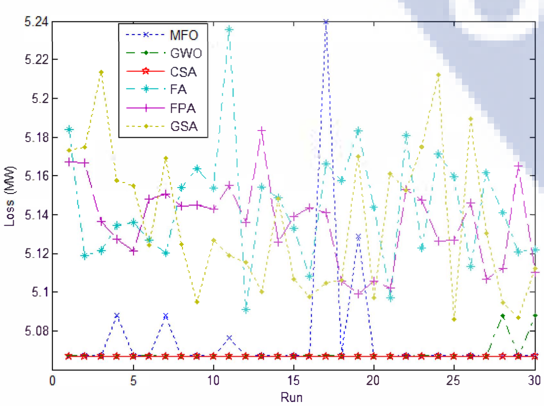

Figure 2. Performance for 30 free running simulations for nature inspired algorithms.

To study the performance of all algorithms in order to find the optimal values of control variables, 30 free running simulations have been carried out which is depicted in Fig. 2. From this figure, it can be seen that the superiority of CSA compared to other algorithms where the CSA is able to perform consistently throughout the 30 simulations. It can be noted also that even GWO and MFO gave the similar minimum losses as shown in Table I, the results obtained by them are varied from the minimum 5.0667 to 5.09 MW.

The convergence graph for all algorithms are shown in Fig. 3. From this figure, all algorithms are converged within 70 iterations except for FPA, where it converged about at 100 iterations. It can be seen also that GWO is the fastest algorithm that converged to the minimum loss.

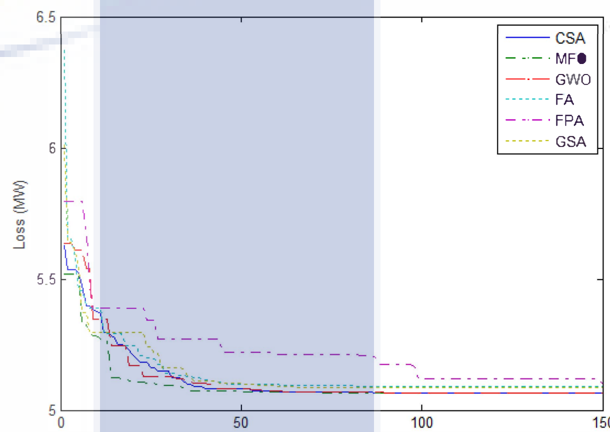

Figure 3. Convergence curve for the best results of nature inspired algorithms.

TABLE II. ORPD RESULTS FOR IEEE-30 BUS SYSTEM

Variables	CSA	MFO	GWO	FA	FPA	GSA
V_1	1.05	1.05	1.05	1.05	1.0498	1.05
V_2	1.0445	1.0446	1.0445	1.0439	1.0449	1.0447
V_5	1.0245	1.0246	1.0246	1.0232	1.0201	1.0247
V_8	1.0264	1.0264	1.0262	1.0264	1.026	1.0268
V_{11}	1.05	1.05	1.05	1.0265	1.0235	1.05
V_{13}	1.05	1.05	1.05	1.0493	1.05	1.05
T_1	1.0374	1.0378	1.0379	1.0019	1.05	1.0073
T_2	0.95	0.95	0.95	0.9939	0.95	0.95021
T_3	0.95	0.95	0.95	0.98696	0.95	0.95411
T_4	0.95107	0.95101	0.95095	0.96491	0.95	0.95768
Q_3	9.081	9.004	8.9915	4.5068	19.375	15.026
Q_{10}	31.64	31.775	31.8	26.563	36	18.701
Q_{24}	11.028	10.923	10.965	12.476	9.2345	12.939
Loss (MW)	**5.0667**	**5.0667**	**5.0667**	**5.0908**	**5.0992**	**5.0861**

TABLE III. RESULTS OF VOLTAGE MAGNITUDE FOR IEEE-30 BUS SYSTEM

Bus No	Before ORPD	CSA	MFO	GWO	FA	FPA	GSA
1	1.06	1.05	1.05	1.05	1.05	1.05	1.05
2	1.05	1.04	1.04	1.04	1.04	1.04	1.04
3	1.04	1.04	1.04	1.04	1.04	1.04	1.04
4	1.03	1.03	1.03	1.03	1.03	1.03	1.03
5	1.01	1.02	1.02	1.02	1.02	1.02	1.02
6	1.02	1.03	1.03	1.03	1.03	1.03	1.03
7	1.01	1.02	1.02	1.02	1.02	1.02	1.02
8	1.01	1.03	1.03	1.03	1.03	1.03	1.03
9	1.00	1.04	1.04	1.04	1.03	1.02	1.04
10	0.99	1.05	1.05	1.05	1.03	1.05	1.04
11	1.08	1.05	1.05	1.05	1.03	1.02	1.05
12	1.02	1.05	1.05	1.05	1.04	1.05	1.05
13	1.07	1.05	1.05	1.05	1.05	1.05	1.05
14	1.00	1.04	1.04	1.04	1.02	1.04	1.04
15	1.00	1.04	1.04	1.04	1.02	1.03	1.03
16	1.00	1.04	1.04	1.04	1.03	1.04	1.04
17	0.99	1.04	1.04	1.04	1.03	1.04	1.04
18	0.98	1.03	1.03	1.03	1.01	1.03	1.03
19	0.98	1.03	1.03	1.03	1.01	1.02	1.02
20	0.98	1.03	1.03	1.03	1.02	1.03	1.03
21	0.98	1.04	1.04	1.04	1.02	1.04	1.04
22	0.98	1.04	1.04	1.04	1.03	1.04	1.04
23	0.98	1.03	1.03	1.03	1.02	1.03	1.03
24	0.96	1.04	1.04	1.04	1.02	1.03	1.04
25	0.95	1.04	1.04	1.04	1.03	1.04	1.04
26	0.93	1.03	1.03	1.03	1.01	1.02	1.02
27	0.94	1.05	1.05	1.05	1.04	1.05	1.05
28	1.02	1.02	1.02	1.02	1.02	1.02	1.02
29	0.92	1.03	1.03	1.03	1.02	1.03	1.03
30	0.91	1.02	1.02	1.02	1.00	1.02	1.01
Loss (MW)	**5.6630**	**5.0667**	**5.0667**	**5.0667**	**5.0908**	**5.0992**	**5.0861**

The study is further analyzed with the effects of control variables setup for IEEE-30 bus system. Even though that the minimum loss is obtained, the voltage of each bus supposed to be operated within their limits where for this study, the load buses must be operated at ± 10% whereas for generator buses must be operated as specified in Table1 which is ± 5%. The details voltage magnitude results for all buses are tabulated in Table III. It can be seen that all voltage magnitudes are operated at the specified limits for all algorithms and improved from the base case which is before the ORPD solution. These can be referred at buses 1, 3, 5, 8, 11 and 13.

VI. CONCLUSION

This paper has proposed a nature inspired algorithm, CSA in solving the ORPD problem. The effectiveness of CSA was exhibited using IEEE 30-bus system as a case study. Simulation results showed that CSA is better compared to other identified algorithms in terms of the consistency of obtaining the minimum loss of the system. The implementation of CSA into other objective functions such as voltage deviation as well as including the practical constraints related to generating units will be proposed in the near future.

ACKNOWLEDGMENT

This research study is supported by Universiti Malaysia Pahang under UMP Research Grant #150362.

REFERENCES

[1] K. Ayan and U. Kılıç, "Artificial bee colony algorithm solution for optimal reactive power flow," Applied Soft Computing, vol. 12, pp. 1477-1482, 2012.

[2] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, “Using the gray wolf optimizer for solving optimal reactive power dispatch problem,” Applied Soft Computing, vol. 32, no. 0, pp. 286–292, 2015.

[3] G. Chen, L. Liu, Z. Zhang, S. Huang, "Optimal reactive power dispatch by improved GSA-based algorithm with the novel strategies to handle constraints," Applied Soft Computing, vol. 50, pp. 58-70, 2017.

[4] A. Mukherjee and V. Mukherjee, "Solution of optimal reactive power dispatch by chaotic krill herd algorithm," IET Generation, Transmission & Distribution, vol. 9, pp. 2351-2362, 2015.

[5] S. B. Raha, T. Som, K. K. Mandal, N. Chakraborty, "Cuckoo search algorithm based optimal reactive power dispatch," in Proceedings of The 2014 International Conference on Control, Instrumentation, Energy and Communication (CIEC), 2014, pp. 412-416.

[6] Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J., "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," IEEE Transactions on Power Systems, vol.26, no.1, pp.12-19, Feb. 2011.

[7] X. S. Yang and D. Suash, "Cuckoo Search via Levy flights," in 2009 World Congress on Nature & Biologically Inspired Computing (NaBIC), 2009, pp. 210-214.

[8] M. A. Mellal, E. J. Williams, “Cuckoo optimization algorithm with penalty function for combined heat and power economic dispatch problem,” Energy, vol. 93, pp. 1711-1718, 2015.

[9] M. A. Mellal, E. J Williams, “Parameter optimization of advanced machining processes using cuckoo optimization algorithm and hoopoe heuristic,” Journal of Intelligent Manufacturing, vol. 27, no. 5, pp. 927-942, 2016.

[10] M. A. Mellal, E. J. Williams, “Cuckoo optimization algorithm for unit production cost in multi-pass turning operations,” The International Journal of Advanced Manufacturing Technology, vo. 76, no. (1-4), pp. 647-656, 2015.

[11] M. A. Mellal, S. Adjerid, E. J. Williams, “Optimal selection of obselete tools in manufacturing systems using cuckoo optimization algorithm,” Chemical Engineering Transactions, vol 33, no. 1, pp. 355-360, 2013.

[12] M. A. Mellal, S. Adjerid, E. J. Williams, D. Benazzouz, “Optimal replacement policy for obselete components using cuckoo optimization algorithm based-approach: dependability context,” Journal of Scientific & Industrial Research, vol. 71, no. 11, pp.715-721, 2012.

[13] S. Mirjalili, “Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm,” Knowledge-Based Syst., vol. 89, pp. 228–249, Nov. 2015.

[14] X-S Yang, “Firefly algorithm, stochastic test functions and design optimisation,” International Journal of Bio-Inspired Computation, vol. 2, no. 2, pp. 78-84, 2010.

[15] X-S Yang, “Flower pollination algorithm for global optimization,” International Conference on Unconventional Computing and Natural Computation, pp. 240-249, 2012.

Rebecca Ng Shin Mei [1,*], Mohd Herwan Sulaiman[1], Zuriani Mustaffa[2]

J. Electrical Systems "*Special issue AMPE2015*"

Selected paper

Ant lion optimizer for optimal reactive power dispatch solution

Journal of Electrical Systems

This paper proposes the application of the recent meta-heuristic method namely Ant Lion Optimizer (ALO) in solving Optimal Reactive Power Dispatch (ORPD) problem. The objective is to minimize the transmission losses by finding the best combination of control variables including generator voltages, transformer tap ratios and reactive compensation devices. In order to show the effectiveness of ALO in solving ORPD, IEEE 30-bus system is utilized. The comparison with other methods also reported in this paper.

Keywords: Optimal reactive power dispatch; Ant lion optimizer; meta-heuristic method.

1. Introduction

Optimal reactive power dispatch (ORPD) is a nonlinear optimization problem in power system which involving discrete and continuous control variables meanwhile satisfying both equality as well as inequality constraints. ORPD is a sub problem of optimal power flow (OPF) calculations which identifies the controllable variables besides minimizes transmission losses and other objective functions. Since transformer tap ratios and outputs of shunt capacitors have discrete nature, whereas, on the other hand, reactive power output of generators and static VAR compensators, bus voltage magnitude and angles are continuous variables. The ORPD therefore can be formulated as a large scale mixed integer nonlinear programming (MINLP) model [1-4]. Undeniably, ORPD plays an important role in securing both electricity and economic operation of power system.

Various techniques on ORPD have been reported in literature. According to [1, 5-9], classical methods including linear and nonlinear programming (LP & NLP), quadratic programming (QP), gradient method, interior point method as well as Newton method have been carried out to solve ORPD problem. Nevertheless, latter development in meta-heuristic methods can yield a better outcome in overcoming ORPD problem compared with classical conventional method. Furthermore, a numerous noticeable search techniques have been implemented for solving ORPD problem such as Genetic Algorithm (GA), Evolutionary Programming (EP), Evolutionary Strategy (ES), Particle Swarm Optimization (PSO) and Tabu Search (TS). However, they are not efficient in solving optimization problems with discrete nature although they are excellent in producing global optimum as well as in overcoming non-convex and discontinuous objective functions [2]. Hence, meta-heuristic methods have been developed to solve ORPD such as Artificial Bee Colony (ABC) [2], Particle Swarm Optimization (PSO) [3, 10], Differential Evolution (DE) [4], Harmony Search Algorithm (HSA) [11], Gray Wolf Optimizer (GWO) [12] and many more.

In this paper, application of Ant Lion Optimizer (ALO) [13] has implemented in solving ORPD problem. The rest of this paper is organized as follows: Section 2 presents the notation used throughout the paper. Section 3 discusses the problem formulation of ORPD

[*] Corresponding author: Rebecca Ng Shin Mei, Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang Malaysia, E-mail: rebeccamei527@gmail.com

[1] Faculty of Electrical & Electronics Engineering, Universiti Malaysia Pahang, 26600 Pekan, Pahang Malaysia

[2] Faculty of Computer System & Software Engineering, Universiti Malaysia Pahang, 26300 Gambang, Pahang Malaysia

followed by a brief description of ALO. Section 4 introduces on the case study as well as the simulation results and discussion. Last but not least, the conclusion is stated in Section 5.

2. Notation

The notation used throughout the paper is stated below.

Constants:

x	vector of dependent variables
u	vector of control variables
Nl	number of transmission lines
V_i	voltage at load bus-i
V_j	voltage at load bus-j
P_{Di}	active load demand
Q_{Di}	reactive load demand
G_{ij}	conductance between bus-i and bus-j
B_{ij}	susceptance between bus-i and bus-j
P_{Gi}	real power generation
Q_{Gi}	reactive power generation
V_{Gi}	generation of bus voltage
T_i	transformer tap setting
Q_{Ci}	reactive compensators
N_G	number of generators
N_T	number of transformers
N_C	number of shunt compensators
cumsum	cummulative sum
$r(t)$	stochastic function
rand	random number
X_i^t	min-max normalization
a_i	minimum of random walk of i-th variable
c_i^t	minimum of i-th variable at t-th iteration
d_i^t	maximum of i-th variable at t-th iteration
c_i	minimum of all variables for i-th ant
d_i	maximum of all variables for i-th ant
c^t	minimum of all variables at t-th iteration
d^t	maximum of all variables at t-th iteration
I	ratio
$Antlion_j^t$	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
Ant_i^t	position of the selected j-th ant at t-th iteration
R_A^t	random walk around the antlion selected by the roulette wheel at t-th iteration
R_E^t	random walk around the elite at t-th iteration

3. Problem formulation

3.1. Objective function

The objective function of ORPD is to determine the minimum system transmission losses and the smallest voltage deviation on load busses concurrently satisfying both the equality as well as inequality constraints. The ORPD problem can be formulated as follows:

Minimize $f(x,u)$
Subjected to
$g(x,u)=0$
$h(x,u)\le 0$ (1)
where function $f(x,u)$ is the objective function, $g(x,u)=0$ is the equality constraint which is the power flow equalities and $h(x,u)\le 0$ is the inequality constraint. Undeniably, transmission losses must be taken into account as it is an economic loss which does not provide any profit. The total transmission loss, *F*, is expressed as follows:

$$F = P_{Loss}(x,u) = \sum_{L=1}^{Nl} P_{Loss} \tag{2}$$

3.2. Constraints

For equality constraint, the total power generation must be equal to the total loads demands and the total real power losses of the system, which can be illustrated as follows:

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

$$Q_{Gi} - Q_{Di} = V_i \sum_{j\in N_i} V_j (B_{ij}\cos\theta_{ij} - G_{ij}\sin\theta_{ij}) \tag{4}$$

There are basically three inequality constraints: generator constraints, transformer tap setting and as well reactive compensators (or shunt VARs). For generator constraints, the real and reactive power generation and generation bus voltage must be within their upper and lower bounds:

$$P_{Gi}^{\min} \le P_{Gi} \le P_{Gi}^{\max}, i = 1,..., N_G \tag{5}$$

$$Q_{Gi}^{\min} \le Q_{Gi} \le Q_{Gi}^{\max}, i = 1,..., N_G \tag{6}$$

$$V_{Gi}^{\min} \le V_{Gi} \le V_{Gi}^{\max}, i = 1,..., N_G \tag{7}$$

The transformer tap setting is limited by their upper and lower bounds as below:

$$T_i^{\min} \le T_i \le T_i^{\max}, i = 1,..., N_T \tag{8}$$

The reactive compensators are restricted within their maximum and minimum limits as below:

$$Q_{ci}^{\min} \le Q_{ci} \le Q_{ci}^{\max}, i = 1,..., N_c \tag{9}$$

3.3. Ant Lion Optimizer (ALO)

Ant Lion Optimizer (ALO) is the latter nature-inspired meta-heuristic method introduced by [13] which mimics the hunting behavior of antlions. ALO is exploited based upon five main stages: random walks of ants, building pits, entrapment of ants, catching preys and lastly rebuilding pits. The steps of ALO can be explained as below. In nature, ants move randomly when searching for food which the random movement of ants can be modeled as follows:

$$X(t) = [0, cumsum(2r(t_1)-1), cumsum(2r(t_2)-1), \ldots, cumsum(2r(t_n)-1)] \tag{10}$$

Stochastic function, $r(t)$ is expressed as below where *rand* is a random number produced within [0,1] uniformly.

$$r(t) = \begin{cases} 1 \; if \;\; rand > 0.5 \\ 0 \; if \;\; rand << 0.5 \end{cases} \tag{11}$$

Random walks of ants: For each optimization, ants will update their locations with random walk. In order to update the positions of ants within the boundary of the search space, equation (10) are normalized using the following equation:

$$X_i^t = \frac{(X_i^t - a_i) \times (d_i - c_i^t)}{(d_i^t - a_i)} + c_i \tag{12}$$

Trapping in antlions' traps: The following equations are applied to express the effect of antlions' traps on random walks of ants.

$$c_i^t = Antlion_j^t + c^t \tag{13}$$

$$d_i^t = Antlion_j^t + d^t \tag{14}$$

Building traps: During optimization, ALO employed roulette wheel operator for choosing antlions based on their fitness as this mechanism gives high chance to the fitter antlions for trapping ants.

Sliding ants against towards antlions: Once antlions realize an ant is in trap, they will shoot the sand outward the middle of the trap. This mechanism slides the trapped ant down to the center of the pit which can be illustrated mathematically as below, where *I* is the ratio.

$$c^t = \frac{c^t}{I} \tag{15}$$

$$d^t = \frac{d^t}{I} \tag{16}$$

Catching preys and rebuilding the traps: Catching preys occurred when ants becomes fitter than it predator. Then, antlion will update its latest location of the hunted ant to improve its opportunity of catching new prey, which this mechanism can be modeled as below:

$$Antlion_j^t = Ant_i^t \quad if\ f(Ant_j^t) > f(Antlion_j^t) \tag{17}$$

Elitism: The movements of all ants are be able to be affected by the fittest antlion which we called it elite during each iteration. Thence, it is assumed that each ant randomly walks around a selected antlion by the roulette wheel and the elite concurrently are modeled as follows:

$$Ant_i^t = \frac{R_A^t + R_E^t}{2} \tag{18}$$

4. Results and discussion

To illustrate the effectiveness of the proposed algorithm in solving ORPD problem, the IEEE 30-bus system is used. This system consists of 6 generators, 41 lines, 4 transformers and 3 capacitor banks as reactive compensation located at buses 3, 10 and 24. The maximum and minimum boundaries for control variables are exhibited in Table 1. The load demand for this study is set to $S = P + j\,Q = 2.832 + j1.262$ p.u.

The best result of ALO is presented in Table 2. For fair comparison, the results presented in [11] are also mapped into the MATPOWER program for load flow assessment. It can be noted that the optimal results obtained by ALO gives the lowest power loss among all the techniques. Comparison ALO with the HSA is about 9.6% loss reduction. It can be seen also that all the optimize variables are within the specified boundaries as shown in Table 1.

Table 1: Limit setting for the variables for IEEE 30-bus system

Variables	Lower limit	Upper limit
Generator Voltges	0.9 p.u	1.1 p.u
Tap setting of transformers	0.95 p.u	1.05 p.u
Capacitor banks	-12 MVar	36 MVar

Table 2: ORPD results of control variables by using HSA, PSO, SGA and ALO

Control device	HSA [11]	PSO [11]	SGA [11]	ALO
V_1	1.0726	1.0313	1.0512	1.1000
V_2	1.0625	1.0114	1.0421	1.0948
V_5	1.0399	1.0221	1.0322	1.0759
V_8	1.0422	1.0031	0.9815	1.0774
V_{11}	1.0318	0.9744	0.9766	1.0761
V_{13}	1.0681	0.9987	1.1000	1.1000
T_1	1.01	0.97	0.95	1.03
T_2	1.00	1.02	0.98	1.00
T_3	0.99	1.01	1.04	1.01
T_4	0.97	0.99	1.02	0.98
Q_1	34	17	12	-1

Q_2	12	13	-10	25
Q_3	10	23	30	11
Loss (MW)	5.1090	5.8815	6.5318	4.616

The performance of ALO is further analysed by performing 30 free running simulations. The performance is exhibited in Figure 1. It can be seen that the results are varied between 4.61 and 4.68 MW which is just about 1.5% deviation for 30 runs. The convergence performance for the best and worst results is depicted in Figure 2.

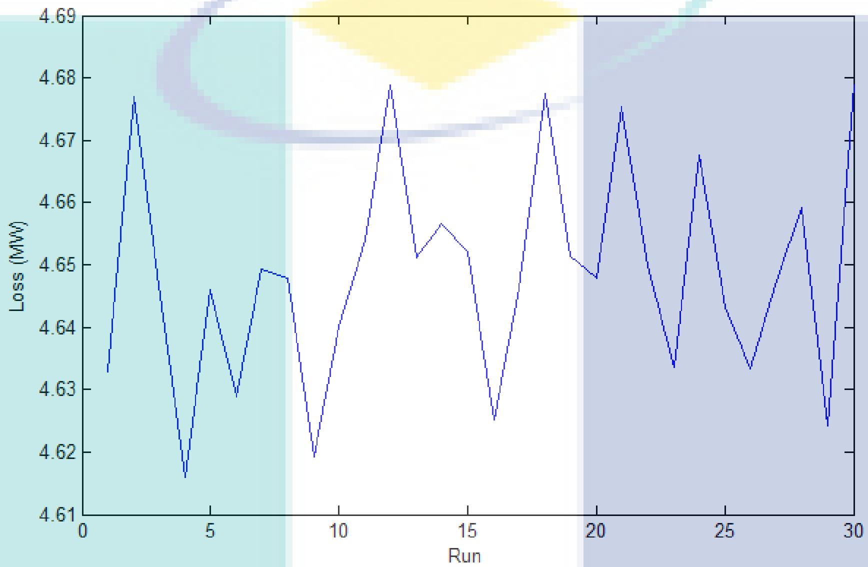

Figure 1: Performance of ALO for 30 free running of simulations

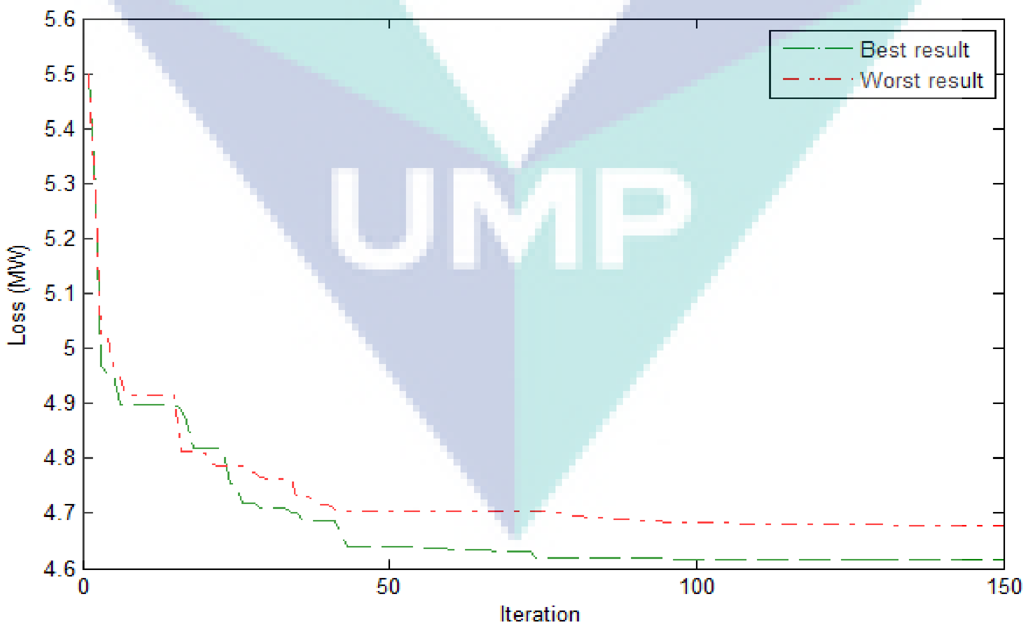

Figure 2: Performance of ALO for the best and worst results

5. Conclusion

A recent meta-heuristic technique namely ant lion optimizer for solving ORPD problem has been presented in this paper. The performance of ALO was evaluated using IEEE 30-bus system. The simulation results show that ALO able to obtain minimum loss compared to other techniques proposed in the literature.

References

[1] M. Varadarajan and K. S. Swarup, "Differential evolution approach for optimal reactive power dispatch," *Applied Soft Computing,* vol. 8, pp. 1549-1561, 9// 2008.

[2] K. Ayan and U. Kılıç, "Artificial bee colony algorithm solution for optimal reactive power flow," *Applied Soft Computing,* vol. 12, pp. 1477-1482, 5// 2012.

[3] B. Zhao, C. X. Guo, and Y. J. Cao, "A multiagent-based particle swarm optimization approach for optimal reactive power dispatch," *Power Systems, IEEE Transactions on,* vol. 20, pp. 1070-1078, 2005.

[4] M. Varadarajan and K. S. Swarup, "Differential evolutionary algorithm for optimal reactive power dispatch," *International Journal of Electrical Power & Energy Systems,* vol. 30, pp. 435-441, 10// 2008.

[5] E. Acha, H. Ambriz-Perez, and C. R. Fuerte-Esquivel, "Advanced transformer control modeling in an optimal power flow using Newton's method," *Power Systems, IEEE Transactions on,* vol. 15, pp. 290-298, 2000.

[6] D. S. Kirschen and H. P. Van Meeteren, "MW/voltage control in a linear programming based optimal power flow," *Power Systems, IEEE Transactions on,* vol. 3, pp. 481-489, 1988.

[7] I. Perez Abril and J. A. G. Quintero, "VAr compensation by sequential quadratic programming," *Power Systems, IEEE Transactions on,* vol. 18, pp. 36-41, 2003.

[8] E. M. Soler, E. N. Asada, and G. R. M. da Costa, "Penalty-Based Nonlinear Solver for Optimal Reactive Power Dispatch With Discrete Controls," *Power Systems, IEEE Transactions on,* vol. 28, pp. 2174-2182, 2013.

[9] P. Subbaraj and P. N. Rajnarayanan, "Optimal reactive power dispatch using self-adaptive real coded genetic algorithm," *Electric Power Systems Research,* vol. 79, pp. 374-381, 2// 2009.

[10] J. G. Vlachogiannis and K. Y. Lee, "A Comparative Study on Particle Swarm Optimization for Optimal Steady-State Performance of Power Systems," *Power Systems, IEEE Transactions on,* vol. 21, pp. 1718-1728, 2006.

[11] A. H. Khazali and M. Kalantar, "Optimal reactive power dispatch based on harmony search algorithm," *International Journal of Electrical Power & Energy Systems,* vol. 33, pp. 684-692, 3// 2011.

[12] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, "Using the gray wolf optimizer for solving optimal reactive power dispatch problem," *Applied Soft Computing,* vol. 32, pp. 286-292, 7// 2015.

[13] S. Mirjalili, "The Ant Lion Optimizer," *Advances in Engineering Software,* vol. 83, pp. 80-98, 5// 2015.

An Application of Moth-Flame Optimization Algorithm for Solving Optimal Reactive Power Dispatch Problem

Mohd Herwan Sulaiman, Zuriani Mustaffa, Omar Aliman, Hamdan Daniyal, Mohd Rusllim Mohamed

***Abstract*—This paper proposes an application of a recent nature inspired optimization technique namely Moth-Flame Optimization (MFO) algorithm in solving the Optimal Reactive Power Dispatch (ORPD) problem. In this paper, loss minimization is used as objective function of ORPD problem where the best combination of control variables such as voltage magnitude, transformer setting and injected MVAR will be obtained by MFO. To show the effectiveness of proposed algorithm, an IEEE 30 bus system is utilized and compared with other algorithms available in literature. The results show that MFO is able to obtain less total system loss than those other algorithms.**

Keywords-Loss minimization, Moth-Flame Optimization, Nature Inspired Algorithms, Optimal Reactive Power Dispatch

I. Introduction

Optimal reactive power dispatch (ORPD) which is partly from optimal power flow (OPF) problems is one of the complex problems in power system engineering. ORPD can be treated as a nonlinear and non-convex problem with equality and inequality constraints that need to be satisfied in order to achieve the objectives in planning and operations of power systems.

One of the objectives for ORPD problems is to minimize the transmission loss. In order to achieve this objective, several main variables need to be controlled and set accordingly such as voltage of generator buses, transformer tap setting as well as the shunt reactive components. It is a nonlinear problems and difficult task since all the controlled variables need to be set simultaneously to achieve the minimum loss. That is why there are massive researches have been done to overcome this problem such as by using classical techniques including sequential quadratic programming [1], non-linear solver with penalty based [2], and Newton techniques [3].

Due to the vast development of recent nature inspired algorithms that have been proved to produce better results in many applications, the implementation of nature based algorithms into solving ORPD problems become popular choice by the researches. The particle swarm optimization (PSO) has been applied in [4] by introducing new philosophy of aging leader and challenger to solve ORPD. The implementation of grey wolf optimizer (GWO) into ORPD also has been done in [5].

Ref. [6] has proposed a solution of ORPD by harmony search algorithm (HSA) which is based on music improvisation of the pitches of instruments to obtain better harmony. Imperialist competitive algorithm (ICA) has been presented in [7] where this technique is based on imperialistic competition in geopolitical between countries. The introduction of invasive weed optimization (IWO) also has been made in order to avoid the local optima trap.

This paper proposes the recent algorithm based on the moth navigation at night which is using the light source for travelling to solve ORPD problem. This algorithm has been proposed by [8]. The organization of this paper is as follows: Section 2 presents the ORPD formulation while brief description of MFO is discussed in Section 3. It is followed by the implementation of MFO into solving ORPD problem in Section 4. Section 5 presents the results and discussion and finally the conclusion is stated in Section 6.

II. Optimal reactive Power Dispatch

ORPD problem can be described as the minimization of function *f*(x, u) subject to the following expressions:

$$g(x,u)=0$$
$$h(x,u)\le 0 \tag{1}$$

where $g(x,u)$ and $h(x,u)$ are the equality and inequality constraints respectively, x is the dependent variables and u is the control variables. The objective of $f(x,u)$ is to minimize the transmission loss system.

The equality constraint equation is the power balanced of load flows which are expressed as follow [6]:

$$\begin{aligned} P_{Gi}-P_{Di} &= V_i\sum V_j\left(G_{ij}\cos\theta_{ij}+B_{ij}\sin\theta_{ij}\right)\\ Q_{Gi}-Q_{Di} &= V_i\sum V_j\left(G_{ij}\cos\theta_{ij}-B_{ij}\sin\theta_{ij}\right)\end{aligned} \tag{2}$$

The inequality constraints are represented in terms of operating constraints such as generators' constraints (upper and lower bound), transformer tap setting as well as reactive elements' upper and lower limits [5]. It is worth to highlight that in this paper that the MATPOWER software package [9] is utilized to obtain total transmission loss by running the load flow program in order to obtain the precise result.

III. Moth-Flame Optimization Algorithm

MFO algorithm is inspired by the moth's special navigation at night. In nature, moths are highly similar to the family of butterflies which has two main milestones in their lifetime which are larvae and adult. Moths have been evolved to fly at night by referring to the moon light by utilizing a mechanism called transverse orientation for navigation. They fly by maintaining a fixed angle with respect to the moon for travelling in a straight path.

Despite of the transverse orientation, moths usually fly spirally around the lights. Basically they are tricked by the artificial light which is initially, they try to maintain to have similar angle with the light to fly in straight line. However, the effort of moths to maintain a similar angle to the light source which is extremely close compared to the moon will causes a deadly spiral fly path for them [8]. This behavior is depicted in the Fig. 1.

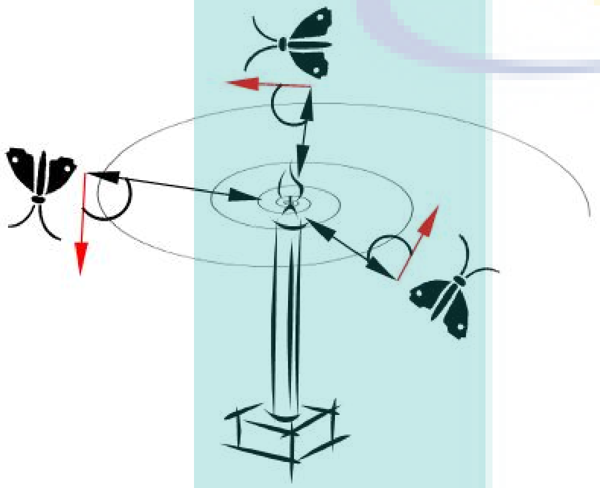

Figure 1. Spiral flying path around close light sources [8].

The development of MFO initially can be described as the following expression:

$$M = \begin{bmatrix} m_{1,1} & m_{1,2} & \cdots & m_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ m_{n,1} & m_{n,2} & \cdots & m_{n,d} \end{bmatrix} \quad (3)$$

where M is a set of moths, n is the number of moths and d is the number of variables (dimension). It is assumed that the candidate of solutions is moths. Other key components in MFO of course are the flames which are constituted also in the form of matrix as follows:

$$F = \begin{bmatrix} F_{1,1} & F_{1,2} & \cdots & F_{1,d} \\ \vdots & \vdots & \cdots & \vdots \\ \vdots & \vdots & \cdots & \vdots \\ F_{n,1} & F_{n,2} & \cdots & F_{n,d} \end{bmatrix} \quad (4)$$

where n and d are the number of moth and dimension respectively. Both (1) and (2) are assumed to store the corresponding fitness values in an array as expressed below:

$$OM = \begin{bmatrix} OM_1 \\ \vdots \\ \vdots \\ OM_n \end{bmatrix} \quad (5)$$

$$OF = \begin{bmatrix} OF_1 \\ \vdots \\ \vdots \\ OF_n \end{bmatrix} \quad (6)$$

It can be seen that both moth and flame are solutions. The difference between them is how to treat and update them for each iteration. In MFO, the moths are actual search agents that move around the search space, whereas flames are the best position of moths so far. Thus, flames can be treated as flags that are dropped by moths when searching the search space. In this mechanism, a moth never loses its best solution [8].

The position of each moth is updated with respect to the flame by using the following expression:

$$M_i = S(M_i, F_j) \quad (7)$$

where M_i indicates the i-th moth, F_j is the j-th flame and S is the spiral function. A logarithmic spiral function is selected as the main update mechanism of moth, such as [8]:

$$S(M_i, F_j) = D_i \cdot e^{bt} \cdot \cos(2\pi t) + F_j \quad (8)$$

Where D_i is the distance of the i-th moth for the j-th flame, b is a constant for defining the shape of the logarithmic spiral and t is a random number between -1 and 1.

To prevent from trapped in local optima, each moth is obliged to update its position using one flame only in eqn. (6). At each iteration and after updating the flames, the flames are sorted based on the fitness values. Then the moths are updating their position with respect to the corresponding flames. Details description of MFO can be obtained in [8].

IV. MFO for ORPD Problem

The implementation of MFO in solving the ORPD problem is by obtaining the optimal values of control variables which is in this paper is the total loss minimization while satisfying al the constrained mentioned in section 2. Initially, the number of moths or search agents and maximum iteration are set. The population (candidate for solution) is constructed in matrix form as depicted in eqn. (3) where the row represents the number of moths and the column represents the number of control variables to be optimized.

To obtain the objective function, each position of moth is mapped into the load flow data and then the load flow program is executed to find the total transmission loss. Once the loss has obtained for respected moth (after updating with the flames such in eqn. (4-6)), the matrix is sorted where the best solution so far is located at the top while the worst result is located at the bottom of the population matrix. If the updated variables are out of bound from the constraints, they are pegged at the minimum or maximum boundaries so that the result obtained is correct. It is also worth to mention that the voltage magnitude for each load bus must be within the specified range, such as ±10%. This is why the MATPOWER load flow program is used in this paper to ensure that the results obtained is valid. The implementation of MFO in solving ORPD is exhibited in Fig. 2.

V. RESULTS AND DISCUSSION

To show the effectiveness of proposed MFO in solving ORPD problem, IEEE 30 bus system is utilized in this paper. The simulation was implemented in MATLAB. The IEEE 30-bus system is based on [5-6] which consists of 13 control variables that need to be optimized. This system consists of six generators, 41 lines, four transformers that located at lines 6-9, 4-12, 9-12 and 27-28 and three shunt reactive elements located at buses 3, 10 and 24. The maximum and minimum boundaries for generators voltages magnitude, transformers tap setting and reactive elements are tabulated in Table 1. The real and reactive load demand for this study is set to 283.2 MW and 126.2 MVar respectively.

TABLE I. BOUNDARIES SETTING FOR CONTROL VARIABLES FOR IEEE 30-BUS SYSTEM

Control Variables	**Lower Bound**	**Upper Bound**
Generator's Voltage Magnitude	0.9 p.u	1.1 p.u
Transformer Tap Setting	0.95 p.u	1.05 p.u
Reactive Shunt Elements	-12 MVar	36 MVar

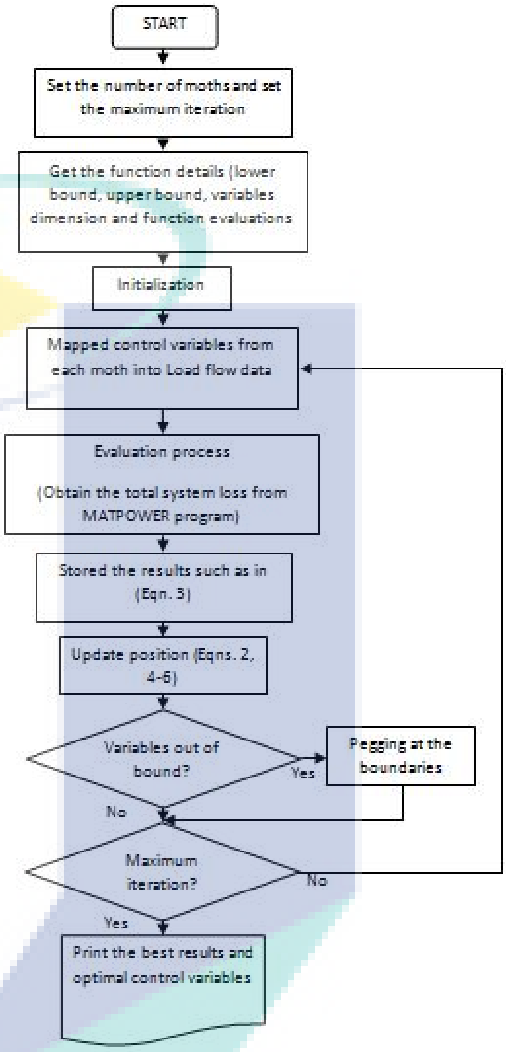

Figure 2. Flow of proposed MFO for solving ORPD problem

In this study, the results in [5-7] are mapped into the same MATPOWER load flow program in order to show fair comparison among the algorithms. Table 2 shows the optimum results of control variables together with the transmission loss obtained from the load flow for selected algorithms. It can be seen that the proposed MFO outperforms all algorithms in obtaining the minimum loss for IEEE 30-bus system. It is also can be noted that all the control variables are within the specified boundaries for all algorithms.

TABLE II. BOUNDARIES SETTING FOR CONTROL VARIABLES FOR IEEE 30-BUS SYSTEM

Variables	MFO #	MFO *	GWO [5]	IWO [7]	ICA [7]	HSA [6]
V_1	1.1	1.1	1.1	1.0697	1.0785	1.0726
V_2	1.0943	1.0946	1.096149	1.0604	1.0694	1.0625
V_5	1.0752	1.0756	1.080036	1.0369	1.047	1.0399
V_8	1.077	1.0772	1.080444	1.0386	1.0471	1.0422
V_{11}	1.0696	1.0868	1.093452	1.0297	1.0349	1.0318
V_{13}	1.1	1.1	1.1	1.0557	1.0711	1.0681
T_1	1.05	1.0411	1.04	1.05	1.08	1.01
T_2	0.95	0.95	0.95	0.96	0.95	1
T_3	0.95	0.96	0.95	0.97	1	0.99
T_4	0.96	0.96	0.95	0.97	0.97	0.97
Q_1	7	7	12	8	−6	34
Q_2	36	31	30	35	36	12
Q_3	10	10	8	11	11	10
Loss (MW)	**4.5869**	**4.5876**	**4.5984**	**4.92**	**4.849**	**5.109**

No. of Search Agent = 20, * No. of Search Agent = 30

In this table also can be seen that two best results of MFO have been presented where the different number of search agents viz. 20 and 30 is exhibited at the first and second column of the table respectively. From the simulations that have been conducted, it can be said that the 20 search agents is adequate to obtain the best results for this system. Fig. 3 shows the performance in terms of loss (MW) versus iterations for various numbers of search agents of MFO. The results depicted in this figure are only the best results of each number of search agents among 30 free running simulations are selected. It can be noted that the result of 20 search agents is slightly better compared to others and adequate to obtain good combination of control variables of ORPD within 150 iterations.

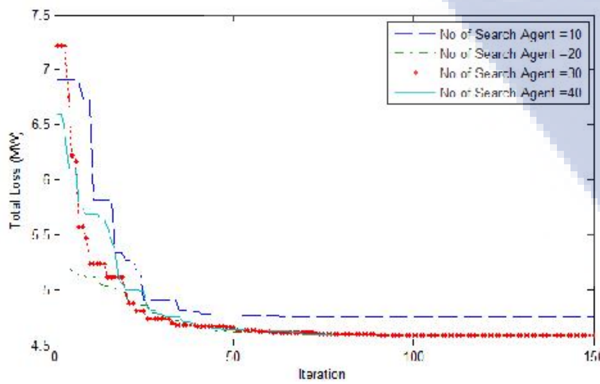

Figure 3. Performance for various numbers of agents of MFO

Fig. 4 show the performance of MFO for 30 free running of simulations for 20 and 30 search agents. It can be seen that it is adequate to obtain a good results by using 20 search agents. It also can be seen that the consistency of 20 search agents compared to 30 search agents of MFO for this test system where the range is between 4.5 to 4.8 MW for 20 search agents compared to between 4.5 to 5.3 ranges for 30 search agents. Nevertheless, both search agents are still the best compared to others as shown in Table 2.

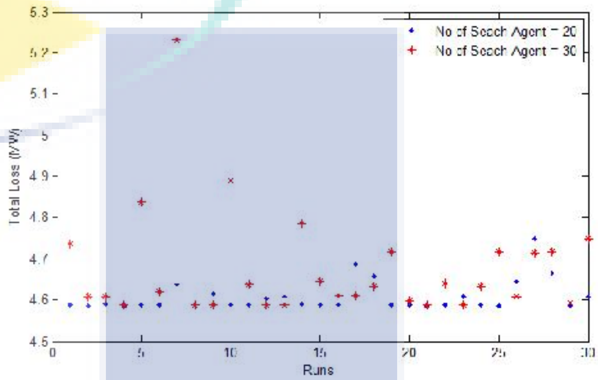

Figure 4. Performance for 20 and 30 earch agents for 30 free running simulations.

VI. CONCLUSION

This paper has proposed a recent nature inspired computing algorithm, Moth-Flame Optimization algorithm in solving ORPD problem. The effectiveness of MFO was demonstrated using IEEE 30-bus system. Simulation results showed that MFO is better compared to other selected algorithms in terms of finding the minimum transmission loss. The implementation of MFO into other objective function such as voltage deviation as well as including the practical constraints related to generating units will be proposed in the near future.

ACKNOWLEDGMENT

This research study is supported by Universiti Malaysia Pahang under UMP Research Grant #150362.

REFERENCES

[1] I. P. Abril and J. A. G. Quintero, “VAR compensation by sequencial quadratic programming,” IEEE Transactions on Power Systems, vol. 18, Dec. 2003, pp. 36-41.

[2] E. M. Soler, E. N. Asada and R. M. da Costa, “Penalty-based nonlinear solver for optimal reactive power dispatch with discrete controls,” IEEE Transacations on Power Systems, vol. 28, no. 3. August 2013, pp. 2174-2182.

[3] E. Acha, H. Ambriz-Perez and C. R. Fuerte-Esquivel, “Advanced Transformer Control Modeling in an Optimal Power Flow Using Newton’s Method,” IEEE Transactions on power Systems, vol. 15, no. 1 February 2000, pp. 290-298.

[4] R. P. Singh, V. Mukherjee, and S. P. Ghoshal, “Optimal reactive power dispatch by particle swarm optimization with an aging leader

and challengers," Applied Soft Computing, vol. 29, pp. 298–309, Apr. 2015.

[5] M. H. Sulaiman, Z. Mustaffa, M. R. Mohamed, and O. Aliman, "Using the gray wolf optimizer for solving optimal reactive power dispatch problem," Applied Soft Computing, vol. 32, no. 0, pp. 286–292, 2015.

[6] 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, Mar. 2011.

[7] 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, Sep. 2014

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

[9] Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J., "MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education," IEEE Transactions on Power Systems, vol.26, no.1, pp.12-19, Feb. 2011.