DEVELOPMENT OF MONARCH BUTTERFLY OPTIMIZATION ALGORITHM FOR ECONOMIC LOAD DISPATCH SOLUTION

By

Sunday Oyong, OBI

June, 2018

DEVELOPMENT OF MONARCH BUTTERFLY OPTIMIZATION ALGORITHM FOR ECONOMIC LOAD DISPATCH SOLUTION

BY

OBI, Sunday Oyong

P15EGEE8053

sunday.obi@nerc.gov.ng

SUPERVISORY COMMITTEE

Dr. Y. Jibril (Chairman)

Prof. Boyi Jimoh (Member)

A Dissertation Submitted to the Department of Electrical Engineering, Ahmadu Bello University, Zaria, in Partial fulfillment of the requirements for the Award of Master of Science (M.Sc.) Degree in Electrical Engineering

June, 2018

DECLARATION

I, OBI Sunday Oyong hereby declare that the work in this dissertation entitled "Development of Monarch Butterfly Optimization Algorithm for Economic Load Dispatch Solution" has been carried out by me in the Department of Electrical Engineering, Faculty of Engineering, Ahmadu Bello University. The information derived from the literatures has been duly acknowledged in the text and a list of references is provided. No part of this dissertation was previously presented for another degree or diploma at this or any other institution.

OBI Sunday Oyong

_____

Signature & Date

CERTIFICATION

This Dissertation entitled "Development of Monarch Butterfly Optimization Algorithm for Economic Load Dispatch Solution" by OBI Sunday Oyong meets the regulations governing the award of the degree of Master of Science (M.Sc.) in Electrical Engineering (Power Systems) of the Ahmadu Bello University, and is approved for its contribution to the knowledge and literary presentation.

Dr. Jibril Yusuf	_____	_____
(Chairman, Supervisory Committee)	Signature	Date
Professor Boyi Jimoh	_____	_____
(Member, Supervisory Committee)	Signature	Date
Dr. Jibril Yusuf	_____	_____
(Head of Department)	Signature	Date
Professor S.Z. Abubakar	_____	_____
(Dean, School of Postgraduate Studies)	Signature	Date

DEDICATION

This dissertation is dedicated to God Almighty, with whom all things are really possible.

ACKNOWLEDGEMENT

There is no amount of appreciation that would be equal to the invaluable contributions made by my project supervisors, Professor B. Jimoh and Dr. Jibril Yusuf. They took time and pains to read through the project manuscript and provided sound academic guidance before this dissertation could be developed to this level, to them I say thank you Sirs.

I want to use this medium to appreciate every person who has one way or the other contributed to the success of this dissertation. I say a big thank you to former Commissioner of Engineering, Safety and Standards Division who made it possible for me to attend lectures during the course work. Engrs. Abdullahi Bello Mohammed and T. S. G. Wudil deserve a big thank you and appreciation. They made it possible and easy for me to have the time to run this program to date.

I also have obligation to thank a group of vibrant scholars that never stop contributing, reading and correcting the project at each stage of its development. More than the above, they kept giving me the push and encouragement to forge ahead with the M.Sc. program whenever the fire to run the program goes down in me. They are Engr. Tijani Salawudeen, Engr. Abdulrahman Olaniyan, Engr. Abubakar Adamu and Engr. Ekundayo Kehinde. Please, I say thank you so much for the wonderful assistance and encouragement.

Engr. Idris Aminu, you are marvelous for without you it could have been more difficult to secure admission. Thank you for all your effort.

To my wonderful and beautiful wife Benny and the four (4) Kids Stefan, Godschild, Enop and Einstein Obi, I appreciate you for the encouragement at home to work hard for this program, the understanding you showed even when I would leave you for days without being there to show you love yet you never complained I love you all and I say thank you and be blessed by our Father, God Almighty.

ABSTRACT

The generation of electric power using different fuel resources to meet load demand and losses while satisfying various constraints on the system involves high running cost (cost of fuel). Therefore, it requires electric utilities to minimize the cost of production of electric power by planning and dispatching generating units in an economic and efficient manner to meet the system demand. The aim of the proposed research is to develop an efficient unit commitment based economic load dispatch (ELD) method using the monarch butterfly optimization algorithm in MATLAB R2017a software environment. The above aim was achieved by formulating the economic load dispatch method to minimize cost of generation considering the impact of cost function as system constraints using the application of monarch butterfly optimization algorithm as a tool to optimize the cost. MBO method was used to minimize the cost of supply of electric power to meet increasing demand of customers. The method was modelled and the performance was evaluated by applying the model on the three IEEE standards test system (3-unit IEEE test system, 6-unit test system and the 15-unit). Finally, analysis is done, validated and results. For the 3- unit system, the minimized cost obtained by the MBO model was $1,722.4130/hr for power demand of 150MW and $3,561.3973/hr for 300MW power demand. Similarly, for the 6-unit test system, $9,978.9427/hr was obtained for power demand of 700MW while $17,720.085/hr was obtained for power demand of 1400MW. The model also obtained $32,582.8863/hr for a demand of 2,630MW and $22,797.1231/hr for a demand of 5,260MW on the 15-unit test system. The developed model was finally validated by comparing the result of the 15-unit generator with Differential Evolution Particle Swarm Optimization (DEPSO) technique. For the same power demand of 2,630MW, the DEPSO obtained $32,588.81/h. This comparison showed that the application of MBO ELD model performed better than the DEPSO by 0.0157 (≈ 0.02%) percent in terms of generating cost per hour for load demand of 2,630MW with significant reduction in total power loss when compared with the DEPSO result.

TABLE OF CONTENTS

Page number

Declaration i

Certification ii

Dedication iii

Acknowledgement iv

Abstract vi

Table of Contents vii

List of Figures x

List of Tables xi

List of Abbreviation xii

CHAPTER ONE: INTRODUCTION

1.1 Background of the study

1.2 Significance of The Study 3

1.3 Statement of Problem 4

1.4 Aim and Objectives 4

1.5 Methodology 5

1.6 Dissertation Organization 6

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction 7

2.2 Review of Fundamental Concepts 7

2.2.1 Economic load dispatch 7

2.2.1.1 Economic load dispatch neglecting losses 8

2.2.1.2 Economic load dispatch (ELD) with loss 10

2.2.2 Load dispatch cost function formulation 12

2.2.3 System constraints 14

2.2.3.1 Equality constraints (Energy balance equation) 14

2.2.3.2 Inequality constraints (Generating capacity limit constraints) 15

2.2.3.3 Generator constraints 16

2.2.3.4 Voltage constraints 17

2.2.3.5 Running spare capacity constraints 17

2.2.3.6 Transmission line constraints 18

2.2.3.7 Network security constraints 18

2.2.3.8 Ramp rate limit constraint 19

2.2.4 Conventional Methods for Solving Economic Load Dispatch Problem 19

2.2.4.1 Gradient-search method 19

2.2.4.2 Lambda –Iteration Method 21

2.2.4.3 Newton's Method 22

2.2.4.4 Linear Programming 22

2.2.5 Computational Intelligent Methods for ELD Problem 23

2.2.5.1 Particle Swarm Optimization for ELD problem 24

2.2.5.2 Artificial bee colony for ELD problem 25

2.2.6 Monarch Butterfly Optimization 27

2.2.6.1 Monarch Butterfly Optimization 29

2.2.6.2 Adjusting operator 30

2.2.7 IEEE standard test system 33

2.2.7.1 The IEEE 3-unit Generators System 34

2.2.7.2 The IEEE 6-unit Generators System 34

2.2.7.3 The IEEE 15-unit Generators System 35

2.3 **Review of Similar Works** 37

CHAPTER THREE: MATERIALS AND METHOD

3.1 Introduction 44

3.2 Materials 44

3.2.1 MATLAB Simulation Environment 44

3.2.2 IEEE Test System 45

3.3 Methods 46

3.3.1 Monarch Butterfly Optimization (MBO) for ELD problem 47

3.3.2 Economic Load Dispatch Problem Modeling 51

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 Introduction 56

4.2 Simulation Result of the Proposed Model on IEEE 3-Units ELD Test System 56

4.3 Simulation Result of the Proposed Model on IEEE 6-Units ELD Test System 58

4.4 Simulation Result of the Proposed Model on IEEE 15-Units ELD Test System 59

4.5 Comparison 62

CHAPTER FIVE: CONCLUSION AND RECOMMENDATION

5.1 Summary 63

5.2 Limitations 63

5.3 Conclusion 64

5.4 Significant Contribution 65

5.5 Recommendations for Future Research 65

REFERENCES 67

APPENDICES

Appendix A: MATLAB Code for MBO 71

Appendix B: MATLAB Code for MBO Parameter Initialization 75

Appendix C: MATLAB Code for Sorting Population 76

Appendix D: MATLAB Code for ELD Cost Function 77

LIST OF FIGURES

Page number

Figure 3.1: MATLAB Simulation Environment 45

Figure 3.2: Flowchart of MBO (Wang *et al.*, 2015) 50

Figure 3.3: Flow Chart for Implementation of ELD Cost Function 55

Figure 4.1: Cost Minimization Plot on IEEE 3-Unit ELD Test System 56

Figure 4.2: Cost Minimization Plot on the IEEE 6-Unit ELD Test System 58

Figure 4.3 Cost minimization plot on the IEEE 15-Unit ELD Test System 60

LIST OF TABLES

Page number

Table 2.1: IEEE 3-Unit ELD Test System 34

Table 2.2: IEEE 6-Unit ELD Test System 34

Table 2.3: IEEE 15-Unit ELD Test System 35

Table 2.4: IEEE 15-Units Data with Ramp Rate Limits and Prohibited Operating Zones 36

Table 3.1: Parameters of the MBO Used for Simulation 51

Table 4.1: Best Solution Obtained by Proposed MBO Based ELD Algorithm for 3-Unit ELD Test System 57

Table 4.2: Best Solution Obtained by Proposed MBO Based ELD Algorithm for 6-Unit ELD Test System 59

Table 4.3: Best Solution Obtained by Proposed MBO Based ELD Algorithm for 15-Unit ELD Test System 61

Table 4.4: Validation 62

LIST OF ABBRIVIATIONS

Acronyms	Definition
ABC	Artificial Bee Colony
CI	Computational Intelligent (CI) algorithm
DEPSO	Differential Evolution Particle Swarm Optimization
ED	Economic Dispatch
ELD	Economic Load Dispatch
FFA	Fire Fly Algorithm
GENCOs	Generation Companies
IEEE	Institute of Electrical and Electronics Engineers
ISOs	Independent Systems Operators
kV	Kilo Volt
M. Sc.	Master of Science
MATLAB	Matrix Laboratory
MBO	Monarch Butterfly Optimization
MW	Mega Watt
MWH	Mega Watt-Hour
OPF	Optimal power flow (OPF)
PSO	Particle Swarm Optimization

CHAPTER ONE

INTRODUCTION

1.1 Background to the Study

Planning, operation and the control of interconnected power system bring about numerous challenging problems caused by unwanted requirements which are imposed on electric utility companies. This likelihood has brought about the need to attain system planning as well as system operation of higher level and greater sophistication (Alam, *et al.* 2015). The system Operators are highly interested and obligated to maintain and determine the optimal system state, by satisfying many kinds of operational constraints. To solve this class of problems, elaborate studies of power have really been taken on economic dispatch.

The Economic Load Dispatch (ELD) problem pertains to the optimum generation scheduling of available generating units in a power system to minimize the cost of generation subject to system constraints Kaur, *et al.*, (2015). Due to the rapid growth in demand and supply of electricity, electric power system is becoming increasingly bigger and larger every day. Regular electric supply is the greatest requirements for growing industry and other fields of life. With the increasing reliance of industry, agriculture and day-to-day household comfort upon the continuity of electric supply, reliability of power system has become very important.

Furthermore, Babu & Samala, (2013), said Economic Dispatch (ED) of electric power generation involves the schedule of committed generating unit outputs so as to meet the load demand at minimum operating cost while satisfying all units and system equality and inequality constraints. This involves allocation of active power between the units, as the operating cost is insensitive to the reactive loading of a generator (Kamboj, *et al.* 2016; Kumar, *et al.* 2014).

The economic dispatch problem involves the solution of two different problems. The first of these is the unit commitment or pre-dispatch problem wherein it is required to select optimally

out of the available generating sources to operate, to meet the expected load and provide a specified margin of operating reserve over a specified period of time. The second aspect of the economic dispatch is the online economic dispatch wherein it is needed to distribute the load among the generating units actually paralleled with the system in such a manner to minimize the total cost of supplying the minute – to – minute requirements of the system (Mahdi, *et al.* 2017).

In fact, every electric utility is normally under intense pressure and obligation to provide to its customer's certain level and degree of continuity and quality of service (power flow on transmission lines in a specified range). Therefore, economy and other objectives of power system must be properly coordinated in reaching optimal power dispatch. It is required to look for better and realistic planning system to achieve different objectives along with desired quality of power supply and satisfying at the same time various system constraints (Saber and Rahman 2011; Tiwari and Pandit 2016; Wang and Li 2013).

In a practical power system, the power plants are not located at the same distance from the center of loads and their fuel costs are different. Also, under normal operating conditions, the generating capacity is more than the total load demand and losses. Thus, there are many options for scheduling the generation. With large interconnection of electrical networks, energy crisis, and high prices of energy, it is very important to reduce the running charges of electrical energy that is reduce the fuel consumption for meeting a particular demand (Zhang, *et al.* 2012).

In an interconnected power system, the objective is to find the real and reactive power scheduling of each power plant in such a way as to minimize the operating cost. This means that the generators real and reactive powers are allowed to vary within certain limits so as to meet a particular load demand with minimum fuel cost. This is called the optimal power flow (OPF) problem which is used to optimize the power solution of large scale power system. This

is done by minimizing selected objective functions, while maintaining an acceptable system performance in terms of generator capability limits and output of the compensating devices. The objective functions, also known as cost functions may present economic costs, system security or other objectives (Suman, et al. 2016). The cost function associated with economic load dispatch in power system which inspired this research work is presented.

1.2 Significance of the Study

The modern power system around the world has grown in interruption, complexity and demand. The focus has shifted from mere satisfying demand towards enhanced performance, increased customer focus, low cost, reliable and clean power. In this change perspective, scarcity of energy resources, increasing power generation cost, environmental concern necessitates optimal economic dispatch. In reality power stations neither are at equal distances from load nor have similar fuel cost functions. Hence for providing cheaper power, load has to be distributed among various power stations in a way which results in lowest cost for generation. Practical economic dispatch (ED) problems have highly non-linear objective function with rigid equality and inequality constraints. Thus, the need to develop a new optimal approach which will solve ELD problems considering unit commitment while satisfying constraints has becomes important in a deregulated electricity markets. Hence, the needs for this research work.

The transition of many states utility around the globe including Nigeria from a vertically integrated monopoly structure to a deregulated system of electric utility has brought high demand on the operators of these systems as well as the customers. The cost of maintenance, operation and particularly fuel (running) cost is variable and often high. To minimize this cost, utilities have to find an optimal way of reducing the fuel cost of generating units for better

economic dispatch of all the units in the power station required to satisfy the total load in the system. The application of economic load dispatch of generating units of a power plant using the incremental cost methodology in this study would aid the right use of generators in the system to meet all load demand and losses taking into account the equality and inequality constraints of the system.

1.3 Statement of Problem

The problem of economic load dispatch is to find the optimal output of a number of electricity generating facilities, to meet the minimum operating cost, subject to a number of constraints (usually equality and inequality constraints). This problem is nonlinear in nature which is associated with the unit commitment, valve point effect, multi fuels operation and ramp rate. This is usually a difficult optimization problem to solve by using the conventional method due to lots of local minima associated with the ELD problem. Recently, application of computational intelligence paradigm has presented a number of efficient tools for addressing this problem. However, the review of earlier research work in this area failed to consider critical real-life constraints that has impact on minimizing cost. Therefore, this research work is aim at solving ELD problem using practical constraints that has direct impact on the cost by using application of monarch butterfly optimization (MBO) algorithm which is capable of obtaining a more efficient result.

1.4 Aim and Objectives

The aim of the research is to develop an efficient unit commitment based economic load dispatch method using the application of monarch butterfly optimization in MATLAB software environment. In order to achieve the stated aim, the following objectives were employed

I. To formulate the mathematical model of the economic load dispatch (ELD) problem considering the various constraints. This is to establish all the mathematical information needed to perform the optimization.

II. To use the MBO algorithm developed in the ELD mathematical problem model formulated in I.

III. To evaluate and compare the performance of the proposed technique with the work of Sayah & Hamouda, (2013).

1.5 Methodology

The following methodology will be adopted

I. Formulation of the economic load dispatch cost minimization problem
    a. Formulate the constraints and initializing the boundary condition of each constraint
    b. Formulate the objective function based on the economic load dispatch and the initialized constraints in (a)

II. Developing the monarch butterfly optimization algorithm

III. Initialization of algorithm parameters (such as: counter, population, maximum generation, Land1 position, Land2 position, BAR, Peri, Migration ratio and maximum step)
    a. Generate initial position randomly
    b. Perform butterfly migration operator
    c. Perform butterfly adjustment operator
    d. Fitness Evaluation
    e. Divide the butterfly population into two subpopulations (Land1 and Land2)
    f. Generate new positions according to migration and adjustment operator

g. Merge the population in (g) and evaluate the fitness

h. Update until the best solution is obtained.

IV. Obtain and model the IEEE standard ELD test system in MATLAB R2017a.

V. Perform evaluation by applying the developed models in items I and II on item III.

VI. Presentation of result, analysis and validation.

1.6 Dissertation Outline

The general introduction has been presented in chapter one. The rest of chapters in this dissertation is organized as follows: Details of the Review of literatures comprises of the review of fundamental concepts and similar works is presented in chapter two. Chapter three contains the research materials and methods, while chapter four gives details of the results obtained and followed by discussion of such results. Finally, chapter five presents the conclusion and recommendations which gives the summary and provide guidelines for future work. Quoted references and Appendices are also provided at the end of the dissertation.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This section comprises two subsections. In the first subsection, concepts which are fundamental for the success of the proposed research work presented. In the second subsection, some of the related research works done on the area of economic load dispatch problems are reviewed.

2.2 Review of Fundamental Concepts

Fundamental concepts such as optimum load dispatch, cost function formulation, system constraint, Monarch Butterfly Algorithm are presented.

2.2.1 Economic load dispatch

Economic load dispatch involves the solution of two different problems. The first problem is that of the unit commitment or pre-dispatch problem which requires the selection of optimal generating sources out of the available generating units to operate, to meet the expected load demand and provide a given margin of operating reserve over a given period of time. The second problem of economic dispatch is the on-line economic dispatch that requires the distribution of load among the generating units in such a manner as to minimize the total cost of supplying the minute-to-minute requirements of the electrical power system (Subathra, *et al.* 2015).

With large interconnection of electric networks, yet energy crisis in the world is increasing with rise in prices, it is very important to reduce the running costs of electric energy, that is reduce the consumption of fuel for meeting certain load demand. It is pertinent to say that in economic

load dispatch, generations are not fixed but are allowed to take values within certain limits so as to meet a particular load demand with minimum fuel consumption. This means that economic load dispatch problem is really the solution of a large number of load flow problems and choosing the one that is optimal in the sense that it needs minimum cost of generation

It is obvious that total cost of generation is a function of the individual generation of the sources which can take values within certain constraints. The cost of generation is not fixed for a particular load demand but depends on the operating constraints of the sources (Trivedi, *et al.* 2016a). Modern power system has to operate under various operating and network constraints. Therefore, it is best to understand the different constraints before taking up the load dispatching problem.

2.2.1.1 Economic load dispatch neglecting losses

In power system generation, the aim is to maintain an ideal situation, which is transmitting the generated power without loss of power. In this situation, the equation given in (2.1) holds.

$$F = F_T + \lambda\left(P_D + \sum_{n=1}^{N} P_n\right) \tag{2.1}$$

where;

F_T = total generation cost in \$/h and λ is the Lagrange Multiplier.

N is the number of the generating units in the system;

P_D is the real power demand of n^{th} generator in MW

Using lambda iteration by differentiating F with respect to the generation P_n and equating to zero gives the condition for optimal operation of the system (Dasgupta, *et al.* 2016; Shahir, *et al.* 2015).

$$\frac{\partial F}{\partial P_n} = \frac{\partial F_T}{\partial P_n} + \lambda(0-1) \tag{2.2}$$

where;

$$F_T = F_1 + F_2 \ ..., + F_N \tag{2.3}$$

$$\frac{\partial F_T}{\partial P_n} = \frac{dF_n}{dP_n} = \lambda \tag{2.4}$$

Thus, the condition for optimum operation is given as

$$\frac{dF_1}{dP_1} = \frac{dF_2}{dP_2} = ... = \frac{dF_n}{dP_n} \tag{2.5}$$

The incremental production cost of a given plant over a limited range is represented by

$$\frac{dF_n}{dP_n} = F_{nn} / P_n + f_n \tag{2.6}$$

where;

F_{nn} is slope of incremental production cost curve

f_n is intercept of incremental production cost curve and

P_n is intercept of incremental production cost curve and

The active power generation constraints are taken into account while solving the equations which are derived above. If these constraints are violated for any generator it is limited to the corresponding limit and the rest of the load is distributed among the remaining generating units according to the equal incremental cost of production.

2.2.1.2 Economic load dispatch (ELD) with loss

To achieve true ELD, transmission losses must be taken into account. Using B-coefficients method, the transmission losses are expressed using George's formula. The optimal load dispatch problem including transmission losses is defined as (Shahir, *et al.* 2015):

$$\min F_T = \sum_{n=1}^{N} F_n \tag{2.7}$$

Subject to $P_D + P_L - \sum_{n=1}^{N} P_n$

where;

P_L is the total system loss which is assumed to be a function of generation.

Making use of the Lagrange multiplier λ in (2.1), then the auxiliary function is given by

$$F = F_T + \lambda\left(P_D + P_L - \sum_{n=1}^{N} P_n\right) \tag{2.8}$$

The partial differential of this expression when equated to zero gives the condition for optimal Load dispatch i.e.

$$\frac{\partial F}{\partial P_n} = \partial F_T + \lambda\left(\frac{\partial P_L}{\partial P_n} - 1\right) \tag{2.9}$$

$$\frac{dF}{dP_n} + \lambda\left(\frac{\partial P_L}{\partial P_n}\right) = \lambda \tag{2.10}$$

Here, the term $\left(\frac{\partial P_L}{\partial P_n}\right)$ is known as the incremental transmission loss at plant n and λ is known as the incremental cost of received power in $ per MWhr. The above equation is a set of n

equations with (n+1) unknowns i.e. 'n' generations are unknown and λ is unknown. These equations are also known as coordination equations because they coordinate the incremental transmission losses with the incremental cost of production. To solve these equations, the loss formula is expressed in terms of generation as

$$P_L = \sum_m \sum_n P_m B_{mn} P_n \tag{2.11}$$

where;

P_m and P_n are the source loadings,

B_{mn} the transmission loss coefficient

$$\frac{\partial P_L}{\partial P_n} = 2\sum_n B_{mn} P_n \tag{2.12}$$

Also,

$$\frac{dF_n}{dP_n} = F_{nn} P_n + f_n \tag{2.13}$$

Therefore, the coordination equation can be rewritten as

$$F_{nn} P_n + f_n + \lambda \sum_n 2B_{mn} P_m = \lambda \tag{2.14}$$

Solving for P_n gives:

$$P_n = \left(1 - \frac{f_n}{\lambda} \times \sum_{m!=n} 2B_{mn} P_m\right) \Big/ \left(\frac{F_n}{\lambda} + 2B_{mn}\right) \tag{2.15}$$

When transmission losses are included and coordinated, the following points must be kept in mind for economic load dispatch solution (Dasgupta, *et al.* 2016; Shahir, *et al.* 2015).

I. Whereas incremental transmission cost of production of a plant is always positive, the incremental transmission losses can be both positive and negative.

II. The individual generators will operate at different incremental costs of production.

III. The generation with highest positive incremental transmission loss will operate at the lowest incremental cost of production.

2.2.2 Load dispatch cost function formulation

The principal objective of economic load dispatch (ELD) of electric power generation is to schedule (allocate) the generating units so that the load demand is met at minimum operating cost while satisfying certain constraints (equality and inequality constrains). The cost minimization objective of ELD problem is given as follows (Saber and Rahman 2011):

$$F_T = \min \sum_{i=1}^{n} F_i(P_i) \tag{2.16}$$

where;

F_T is the total generation cost in \$/h,

P_i is the real power generation of i^{th} generator in MW,

$F_i(P_i)$ is the generation fuel cost for generating unit with output P_i subject to equality and inequality constraints.

n is the number of the generating units in the system

The cost of a generating unit can be modeled as follows (Wang and Li 2013)

$$F_i(p_i) = a_i p_i^2 + b_i p_i + c_i \tag{2.17}$$

where a_i, b_i and c_i are the cost coefficients of the *ith* generating unit.

Equation (2.17) is called the smooth quadratic function. In reality, the generating units with multi-valve steam turbine have valve-point loading effect. Thus, the cost function contains the following higher-order nonlinearity (Saber and Rahman 2011; Wang and Li 2013):

$$F_i(p_i) = a_i p_i^2 + b_i p_i + c_i + \left| d_i \sin\left(e_i\left(p_i^{\min} - p_i\right)\right)\right| \tag{2.18}$$

where d_i and e_i are valve loading coefficients of the i^{th} generating unit.

The sinusoidal function in equation (2.18) is added to incorporate the valve loading effect. Another realistic representation of generating unit is the multiple fuels. In such a case, the fuel cost function is expressed as the following piecewise quadratic function (Saber and Rahman 2011; Wang and Li 2013):

$$F_i(p_i) = \begin{cases} a_i, p_i^2 + b_{i,1} p_i + c_{i,1} & ip_i \le p_i < p_{i,1} \\ a_{i,2} p_i^2 + b_{i,2} p_i + c_{i,2} \quad if & p_{i,1} \le p_i < p_{i,2} \\ \bullet & \bullet \\ \bullet & \bullet \\ \bullet & \bullet \\ a_{i,k} p_i^2 + b_{i,k} p_i + c_{i,k} \quad if & p_{i,k-1} \le p_i \le p_i^{\max} \end{cases} \tag{2.19}$$

where;

$a_{i,j}, b_{i,j}, c_{i,j}$ are the cost coefficients of the *ith* generating unit for type j

$p_{i,j}$ is the upper bound of the output for type j (j = 1, 2, . . ., k)

$p_i^{\min}$ and $p_i^{\max}$ are the lower bound and upper bound of the output of the *ith* generating unit respectively.

The more practical and accurate ELD problem may consider both valve point loading and multiple fuels thus the cost function can be described as follows (Saber and Rahman 2011; Tiwari and Pandit 2016; Wang and Li 2013):

$$F_i(p_i) = \begin{cases} a_i, p_i^2 + b_{i,1}p_i + c_{i,1} + \left|d_{i,1}\sin\left(e_{i,1}\left(p_i^{\min} - p_i\right)\right)\right| & if \quad p_i \le p_i < p_{i,1} \\ a_{i,2}p_i^2 + b_{i,2}p_i + c_{i,2} + \left|d_{i,2}\sin\left(e_{i,2}\left(p_i^{\min} - p_i\right)\right)\right| & if \quad p_{i,1} \le p_i < p_{i,2} \\ \bullet \quad \bullet \\ \bullet \quad \bullet \\ \bullet \quad \bullet \\ a_{i,k}p_i^2 + b_{i,k}p_i + c_{i,k} + \left|d_{i,k}\sin\left(e_{i,k}\left(p_i^{\min} - p_i\right)\right)\right| & if \quad p_{i,k-1} \le p_i \le p_i^{\max} \end{cases} \tag{2.20}$$

2.2.3 System constraints

There are two types of constraints when considering and dealing with economic load dispatch problems (Kamboj, *et al.* 2016):

I. equality constraints

II. inequality constraints

These constraints are based on the principle of equilibrium between total system loads (PD), total system generation (PG) and total power losses (PL).

2.2.3.1 Equality constraints (Energy balance equation)

For power balance condition, an equality constraint has to be satisfied. The total power generation by all the generating units should be equal or the same as the total load demand plus the total line (transmission) losses in the system (Gautham and Rajamohan 2016; Nawaz, *et al.* 2017).

$$P_D + P_L = \sum_{i=1}^{n} P_i \tag{2.21}$$

where;

P_D is the total demand in MW,

P_L is the transmission loss of the system in MW

Equation (2.21) denotes that the total generation is equal to the total demand when losses are considered.

2.2.3.2 Inequality constraints (Generating capacity limit constraints)

Inequality constraints involves various cases such as generator constraints, voltage constraints, Running spare capacity constraints, transmission line constraints and network security constraints. The power output of each unit (i) must be greater than or equal to the minimum power $\left(P_{G_i}^{\min}\right)$ permitted and also be less than or equal to maximum power $\left(P_{G_i}^{\max}\right)$ permitted on that specified unit. Therefore, the inequality constraint is expressed as (Park, *et al.* 1993):

$$P_{G_{\min}}^{\min} \leq P_{G_i} \leq P_{G_i}^{\max} \tag{2.22}$$

where;

$P_{G_{\min}}^{\min}$ is the minimum output power

P_{G_i} is the output power of each unit *i*

$P_{G_i}^{\max}$ is the maximum output power.

The inequality constraints are classified into Soft and Hard type. The hard type are those constraints which are definite and specific like the tapping range of an on-load tap changing transformer whereas soft type are those constraints which have some flexibility associated with them like the nodal voltages and phase angles between the nodal voltages. Soft inequality constraints have been very efficiently handled by penalty function methods discussed below (Alam *et al.*, 2016; Wang & Li, 2013).

2.2.3.3 Generator constraints

The generation output of each generator should lie between the minimum and the maximum limits (generator limits). The maximum active power generation of a source is limited again by thermal consideration and also minimum power generation is limited by the flame instability of a boiler. If the power output of a generator for optimum operation of the system is less than a pre-specified value $P_i^{\min}$ the unit is not connected to the bus bar because it is not possible to generate that low value of power from the unit (Alam, *et al.* 2016). Therefore, the generator power P cannot be outside the range stated by the inequality given in equation (2.23).

In the same vein, the maximum and minimum reactive power generation of a source is limited. The maximum reactive power is limited because of overheating of rotor and minimum is limited because of the stability limit of the machine.

The inequality constraints for each generator are expressed in terms of active and reactive power, hence (Das and Sengupta 2015)

$$P_i^{\min} \le P_i \le P_i^{\max} \tag{2.23}$$

where;

$P_i^{\min}$ is the minimum power output limit of the i^{th} generator in MW

$P_i^{\max}$ is the maximum power output limit of the i^{th} generator in MW

Similarly, for reactive power limits

$$Q_p^{\min} \le Q_p \le Q_p^{\max} \tag{2.24}$$

where;

Q_p^{min} is the minimum reactive power output limit of the generator in MVar

Q_p^{max} is the maximum reactive power output limit of the generator in MVar

2.2.3.4 Voltage constraints

It is important that the voltage magnitudes and phase angles at different nodes should vary within certain limits. The voltage magnitudes should vary within certain limits otherwise most of the equipment connected to the system will not operate satisfactorily or additional use of voltage regulating devices will make the system uneconomical. The normal operating angle of transmission lies between 30 to 45 degrees for transient stability reasons: therefore, a higher limit is imposed on angle delta. A lower limit of delta assures proper utilization of transmission facility (Shahir, *et al*. 2015).

$$^{\delta} p\ min \leq {}^{\delta}p\ {}^{\delta} \leq p\ max \quad (2.25)$$

where;

$^{\delta}$ is phase angle at the pth node

$$V_i^{min} \leq V \leq V_i^{max} \quad (2.26)$$

V_i^{min} and V_i^{max} are the minimum and maximum generating voltage of unit i respectively.

2.2.3.5 Running spare capacity constraints

These constraints are needed to meet two major conditions which can occur in the system. These two (2) conditions are (Rizwana, *et al.* 2015):

I. the forced outages of one or more alternators on the system and
II. the unexpected load increase on the system.

The total generation should be such that in addition to meeting load demand and losses a minimum spare capacity should be available that is (Dasgupta, *et al.* 2016; Rizwana, *et al.* 2015):

$$G \geq P_p + P_{so} \tag{2.27}$$

where;

G is the total generation;

P_{SO} is some pre-specified power

A system that is well planned should have minimum spare capacity P_{SO}.

2.2.3.6 Transmission line constraints

This constraint looked at the limiting of the flow of active and reactive power through transmission line circuit due to thermal capability of the circuit and it is expressed as (Dasgupta, *et al.* 2016; Shahir, *et al.* 2015):

$$C_p \leq C_p^{\max} \tag{2.28}$$

where;

$C_p^{\max}$ is the maximum loading capacity of the line.

2.2.3.7 Network security constraints

If a system operates satisfactorily and there is an outage which may be planned or unplanned, it is natural that some of the constraints of the system will be violated. The complexity of the constraints (for instance number of constraints) is increased when a large system is involved.

The natures of these constraints are same as voltage and transmission line constraints (Dasgupta *et al.*, 2016).

2.2.3.8 Ramp rate limit constraints

In this case, the range of operations of all the online units is restricted by their corresponding up-ramp and down-ramp rate limits as follows (Dasgupta, *et al.* 2016; Rizwana, *et al.* 2015):

$$P_i - P_i^0 \leq UR_i \tag{2.29}$$

$$P_i^0 - P_i \leq DR_i \tag{2.30}$$

where;

P_i^0 and P_i are the previous and current power output of the ith generating unit,

DR_i and UR_i are the down-ramp and up-ramp rate of the ith generating unit, respectively.

2.2.4 Conventional Methods for Solving Economic Load Dispatch Problem

In the past few decades, the ELD problems have been addressed using the conventional approach which includes the Gradient Search method, Lambda iteration method and Newtonian method. The procedural implementation of these methods is discussed as follows.

2.2.4.1 Gradient-search method

The gradient search method is based on the principle that the minimum of a function, *f(x)*, can be found by a series of steps in the downward direction. From any starting point, x^0, the direction of "steepest descent" can be found by noting that the gradient *f*, always points in the direction of maximum ascent (Dasgupta, *et al.* 2016; Rizwana, *et al.* 2015; Wood and Wollenberg 2015):

$$\nabla f = \begin{bmatrix} \frac{\partial f}{\partial x_1} \\ \cdot \\ \cdot \\ \cdot \\ \frac{\partial f}{\partial x_n} \end{bmatrix} \tag{2.31}$$

Then we should go from x^0 to x^1 using (Wood and Wollenberg 2015):

$$X^1 = X^0 - \nabla f \alpha \tag{2.32}$$

where;

α is a scalar which ensure convergence and the best value of α must be determined experimentally.

To solve the economic load dispatch problem which involves minimizing the objective function and keeping the equality constraints, we must apply the gradient technique directly to the Lagrange function which is (Wood and Wollenberg 2015)

$$\Im = \sum_{i=1}^{N} F_i(P_i) + \lambda\left(P_{load} - \sum_{i=1}^{N} P_i\right) \tag{2.33}$$

The function $\left(P_{load} - \sum_{i=1}^{N} P_i\right)$ is a constraint function which may affect the objective of the ELD problem. The gradient of the function given in equation (2.34) is given as follows (Wood and Wollenberg 2015):

$$\nabla \Im = \begin{bmatrix} \frac{\partial \Im}{\partial P1} \\ \bullet \\ \bullet \\ \bullet \\ \frac{\partial \Im}{\partial Pn} \end{bmatrix} \tag{2.34}$$

The economic dispatch algorithm requires a starting point of λ and an initial values for P_1, $P2$, and P_3 .The gradient for $\Im$ is calculated as above and the new values of λ ,P_1, P_2 and P_3 are found using (Hosseinnezhad and Babaei 2013):

$$X^1 = X^0 - (\nabla \Im)\alpha \tag{2.35}$$

where X is a vector defined as:

$$X = \begin{bmatrix} P_1 \\ P_2 \\ \bullet \\ \bullet \\ \lambda \end{bmatrix} \tag{2.36}$$

2.2.4.2 Lambda –Iteration Method

The Lambda iteration method is performed by the introduction of a lambda (λ) called the Lagrange multiplier. Since all the inequality constraints must be satisfied in each trial, the equations are solved by the iterative method. The step by step approach of implementing the Lambda-Iteration is as follows (Wood & Wollenberg, 2015):

I. Assumed suitable value of this value should be more than the largest intercept of the incremental cost characteristic of the various generators.

II. Compute the individual generations

III. Check if the equality constraint is satisfied i.e.

$$Pd = \sum_{n=1}^{n} Pn \tag{2.37}$$

where;

P_d is Power demand by the load and

P_n is Real power generated

IV. If not, make the second guess λ and repeat above steps I-III

2.2.4.3 Newton's Method

Newton's method provides a vector function which tries to solve the economic dispatch by observing that the aim is to always drive the objective function towards zero. (Dasgupta, *et al.* 2016; Ghasemi 2012a; Rizwana, *et al.* 2015)

$$\nabla \psi x = 0 \tag{2.38}$$

Since this is a vector function, the problem can be formulated as one of finding the correction that exactly drives the gradient to zero (i.e. to a vector, all of whose elements are zero). Suppose we wish to drive the function g(x) to zero. (Jabr, *et al.* 2000) The function g is a vector and the unknown, x are also vectors.

$$\Delta X = -\left[g(x)\right]^{-1} g(x) \tag{2.39}$$

Now, if the g function is the gradient vector $\nabla \Psi$ x, this gives:

$$\Delta X = -\left[\frac{\partial}{\partial x}\nabla \psi_x\right]^{-1} \Delta \psi \tag{2.40}$$

For the economic load dispatch problem, this takes the form:

$$\psi = \sum_{i=1}^{N} F_i(P_i) + \lambda\left(P_{load} - \sum_{i=1}^{N} P_i\right) \tag{2.41}$$

The vector $\nabla \psi x$ is a Jacobean matrix which now has second order derivatives and is called Hessian matrix. Generally, Newton's method will solve for the correction that is much closer to the minimum generation cost in one cost in one step than would the gradient method (Wood and Wollenberg 2015).

2.2.4.4 Linear Programming

The Linear programming (LP) is a general optimization technique which has been widely employed for solving optimization problem of linear objective function and linear equality or inequality constrains. A typical example

e of this linear objective function is given as follows (Dasgupta, *et al.* 2016; Rizwana, *et al.* 2015)

$$f(x_1, x_2, ...x_n) = c_1x_1 + c_2x_2 + ...c_nx_n + d \tag{2.42}$$

The idea of linear programming is to find a point in the optimization surface where the objective function has the smallest (or largest) value. Such points may not exist, but if they do, searching through the optimization surface vertices is guaranteed to find at least one of them. The general form of linear programming problem is as follows (Jabr, *et al.* 2000):

$$\begin{aligned} &\textit{Maximize} \quad Z = C^T X \\ &\textit{Subject to} \quad AX \leq b \end{aligned} \tag{2.43}$$

X is called the decision variables, which represents the vector of variables whose optimized values is to be determined. While, C and b are vectors of (known) coefficients, A is a (known) matrix of coefficients. The expression to be maximized or minimized is called the objective function (Z in this case). The equation $AX \leq b$ is the constraint over which the objective function is to be optimized. (Jabr, *et al.* 2000)

All previous methods used where not producing optimal solutions because they require very high computational time and involves numerous iterative calculations, hence the need to look for alternative methods (such as computational intelligent) for solving ELD problems.

2.2.5 Computational Intelligent Methods for ELD Problem

Over the years, researchers have sought for a means of addressing some complex real world problems using the cooperative and adaptive behaviors of biological systems. This subsequently led to the development of Computational Intelligent (CI) algorithm such as Particle Swarm Optimization (PSO), Fire Fly Algorithm (FFA), Artificial Bee Colony Algorithm and a host of others. Researchers in the area of ELD have employed these CI algorithms to address the ELD problem with efficient and promising results.

2.2.5.1 Particle Swarm Optimization for ELD problem

Particle swarm optimization (PSO) is an optimization algorithm which was inspired by intelligent behaviors of flocks of birds and school of fish which was proposed by Eberhart in 1995 (Eberhart and Kennedy 1995). The PSO algorithm is a procedure that is initialized by a set of randomly generated candidate solutions and the search for the better particles (solution) is performed iteratively. Assuming the population of the particles is represented as *N* and the dimension with which the particle is to swarm is represented as *D*, the initial position of the particles is generated randomly as a set of vectors as follows (Babaoˇglu, *et al*. 2012; Niu, *et al*. 2012).:

$$X_i^t = \left[x_{N,1}, x_{N,2} ... x_{N,D}\right] \tag{2.44}$$

And the velocity vector of the particles can be formulated as follows:

$$V_i^t = \left[v_{N,1}, v_{N,2} ... v_{N,D}\right] \tag{2.45}$$

This velocity vector is updated as follows (Chen and Li 2007):

$$V_i^{k+1} = wV_i^k + c_1 rand_1 \times \left(Pbest_i^k - X_i^k\right) + c_2 rand_2 \times \left(Gbest_i^k - X_i^k\right) \tag{2.46}$$

where;

V_i^k is the velocity of individual at each iteration;

w is weight parameter;

c_1, c_2 are weight factors;

$rand_1, rand_2$ are random numbers between 0 and 1;

X_i^k is the position of individual at each iteration;

$Pbest_i^k$ is the best position of individual until iteration;

$Gbest_i^k$ is the best position of the group until iteration

In order to update the position of each particle, the position vector of the current particle is updated as follows:

$$X_i^{k+1} = X_i^k + V_i^{k+1} \tag{2.47}$$

The pseudo code implementation of the particle swarm optimization algorithm is given as follows (Sharma and Moses 2016):

1: Generate the initial swarm by randomly generating the position and velocity for each particle ;
2: Evaluate the fitness of each particle;
3: **repeat**
4: **for** each particle i **do**
5: Update particle i
6: if $f(\vec{x}_i) < f(\vec{x}_{pbest})$ then
7: $\vec{x}_{pbest} := \vec{x}_i$
8: if $f(\vec{x}_i) < f(\vec{x}_{gbest})$ then
9: $\vec{x}_{gbest} := \vec{x}_i$
10: **end** if
11: end if
12: **error** for
13: **until** the stop criterion is satisfied

2.2.5.2 Artificial bee colony for ELD problem

Artificial Bee Colony (ABC) algorithm is one of the most recently introduced swarm-based algorithms which were inspired by the intelligent foraging behavior of swarm of honey bee. In artificial bee colony algorithm (ABC), the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution (Akay and Karaboga 2012). The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. At the first step, the ABC generates a randomly distributed initial population of N solutions (food

source positions). Each solution is a D-dimensional vector which can be represented as follows (Karaboga, 2005):

$$X = \left[x_{N,1}, x_{N,2} ... x_{N,D}\right] \tag{2.48}$$

After initialization, the population is subject to repeated cycles (*MCN*)

$$C = 1, 2, ..., MCN \tag{2.49}$$

An artificial onlooker bee chooses a food source depending on the probability value associated with that food source p_i using the following equation (Karaboga and Basturk 2007):

$$p_i = \frac{fit_i}{\sum_{n=1}^{N} fit_n} \tag{2.50}$$

where;

fit_i is the fitness function of the current generation '*i*' which is proportional to the nectar amount of the food source in the position *i* and

N is the number of food sources which is equal to the number of employed bees or onlooker bees.

In order to generate a candidate food position from the old one in memory, the following equation is employed (Karaboga 2005; Karaboga and Basturk 2007):

$$v_{i,j} = x_{i,j} + \varphi_{i,j}\left(x_{i,j} - x_{i,k}\right) \tag{2.51}$$

where $i \in \{1, 2, ..., N\}$ and $j \in \{1, 2, ..., D\}$ are chosen randomly and ϕ_{ij} is a random number between $[1, -1]$.

Assuming that the abandoned food sources is x_i and $j \in \{1,2,...,D\}$ then the scout bee discovers a new food source to be replaced with x_i as follows (Karaboga 2005; Karaboga and Basturk 2007):

$$x_i^{t+1} = x_{\min}^t + rand(0,1) \times (x_{\max}^t - x_{\min}^t) \tag{2.52}$$

The pseudo code implementation of the artificial bee colony is given as follows (Karaboga and Basturk 2007):

1: Initialize the population of solutions *xi,j*, *i* = 1. . .*SN*, *j* = 1. . .*Dx*
2: Evaluate the population
3: cycle=1
4: **repeat**
5: Produce new solutions *υi,j* for the employed bees by using (4) and evaluate them
6: Apply selection process based on Deb's method
7: Calculate the probability values *Pi,j* for the solutions *xi,j* by (1)
8: Produce the new solutions *υi,j* for the onlookers from the solutions *xi,j* selected depending on *Pi,j* and evaluate them
9: Apply selection process based on Deb's method
10: Determine the abandoned solution for the scout, if exists, and replace it with a new randomly produced solution *xi,j*by (3)
11: Memorize the best solution achieved so far
12: cycle=cycle+1
13: **until** cycle=MCN

2.2.6 Monarch Butterfly Optimization

The most important factors in metaheuristic algorithms are the exploitation and exploration search mechanisms. A good metaheuristic algorithm has the ability to strike a balance between these two mechanisms, thereby enhancing the solving of low-dimensional and high-dimensional optimization problems. The exploitation mechanism is based on the present knowledge to seek better solutions, while the exploration mechanism is based on fully searching the problem space for an optimal solution. In general, by analyzing the standard MBO algorithm, it is noticed that it has the ability to explore the search space very effectively

and it also has the ability to find the global optimum within a fast speed; however, it has a poor ability to exploit the local search space due to the occasional use of Levy flight by the updating operators, which leads to large steps or moves (Gai-Ge, *et al.* 2015).

The monarch butterfly optimization (MBO) is an optimization algorithm which was inspired by the migration capability of monarch butterfly. In the MBO, the individual monarch butterflies are located in two geographical regions which are southern Canada and the northern USA (Land1) and the Mexico (Land 2). Thus, the positions of the monarch butterflies are updated in two ways. Firstly, the offspring are generated (position updating) by migration operator, which can be adjusted by the migration ratio (Gai-Ge, *et al.* 2015). It is followed by tuning the positions for other butterflies by means of butterfly adjusting operator. In order to keep the population unchanged and minimize fitness evaluations, the sum of the newly generated butterflies in these two ways remains equal to the original population. The migration operator and butterfly adjusting operator helps to determine the direction of the individual butterflies. In order to efficiently develop the MBO, the migration behavior of the monarch butterfly is idealized into the following rules (Gai-Ge, et al. 2015; Wang, et al. 2015) :

I. The monarch butterflies are located in Land 1 or Land 2. Therefore, the butterflies in Land 1 and Land 2 make up the population of the MBO
II. Offspring butterfly individual is generated by migration operator from monarch butterfly in Land 1 or in Land 2.
III. In order to keep a balanced population, an old monarch butterfly will pass away once a child is generated. In the MBO method, this can be performed by replacing its parent with newly generated one if it has better fitness as compared to its parent. On the other hand, the newly generated one is liable to be discarded if it does not exhibit better

fitness with respect to its parent. Under this scenario, the parent is kept intact and undestroyed.

IV. The monarch butterfly individuals with the best fitness move automatically to the next generation, and they cannot be changed by any operators. This can guarantee that the quality or the effectiveness of the monarch butterfly population will never deteriorate with the increment of generations.

2.2.6.1 Migration operator

Since the monarch butterfly migrate from land 1 to land 2 and vice versa, the number of butterflies in land 1 and in land 2 is determined as follows (Gai-Ge, *et al.* 2015; Wang, *et al*. 2016; Wang, *et al.* 2015):

$$X_{l1} = \lceil p \times NP \rceil \times (NP1) \tag{2.53}$$

$$X_{l2} = (NP - NP_1) \times (NP_2) \tag{2.54}$$

where;

NP is the total number of population.

p is the ratio of monarch butterflies in land 1.

The monarch butterflies in land 1 and land 2 are called subpopulation 1 and subpopulation 2 respectively. The migration process is thus expressed as follows (Gai-Ge, *et al.* 2015):

$$x_{i,k}^{t+1} = x_{r1,k}^{t} \tag{2.55}$$

where;

$x_{i,k}^{t+1}$ indicate the kth butterfly of x_i at generation *t+1*

$x_{r1,k}^{t}$ represent the newly generated position of butterfly r_1 at the current time *t*

The monarch butterfly r is randomly generated using the following equation (Gai-Ge, *et al.* 2015).

$$r = rand \times peri \tag{2.56}$$

where;

peri indicates period of migration which is set as 1.2 (i.e. 12 months a year).

The pseudo code description of the migration operator is given as follows (Gai-Ge, *et al*. 2016; Gai-Ge, *et al*. 2015):

Algorithm 1: Migration operator

Begin
for *i=1* to NP_1 (for all monarch butterflies in Subpopulation 1) **do**
for *k=1* to D (all the elements in the *ith* monarch butterfly) **do**
Randomly generate a number *rand* by uniform distribution;
*r=rand*peri*;
if $r \leq p$ **then**
Randomly select a monarch butterfly in Subpopulation 1 (say r_1);
Generate the *kth* element of x_i^{t+1}
else
Randomly select a monarch butterfly in Subpopulation 2 (say r_2);
Generate the *kth* element of x_i^{t+1}
end if
end for k
end for i
End.

2.2.6.2 Adjusting operator

This butterfly adjusting operator is an additional method for updating the position of the monarch butterflies besides the migration operator. It can be implemented simultaneously with the migration operator, making MBO suitable for parallel processing.

The process of MBO adjustment operator can be described as follows. For all the element in the monarch butterfly *j*, if a randomly generated number rand is smaller than or equal to *p*, it can be updated as (Gai-Ge, *et al*. 2016; Gai-Ge, *et al.* 2015):

$$x_{j,k}^{t+1} = x_{best,k}^{t} \tag{2.57}$$

where;

$x_{j,k}^{t+1}$ indicates the kth element of x_j at generation $t+1$.

$x_{best,k}^{t}$ indicates the kth element of x_{best} which is the best monarch butterfly in land 1 and land 2.

Furthermore,

$$x_{j,k}^{t+1} = x_{r3,k}^{t} \tag{2.52}$$

where;

$x_{r3,k}^{t}$ indicates the kth element of x_{r3} that is randomly selected in Land 2. Here $r_3 \in \{1,2,..,NP_2\}$

This butterfly position is updated as follows if rand > BAR (BAR is the butterfly adjusting rate), then:

$$x_{j,k}^{t+1} = x_{j,k}^{t+1} + \alpha \times (dx_k - 0.5) \tag{2.53}$$

The dx_k is the walk step of the monarch butterfly *j* that can be calculated by Levy flight as:

$$dx_k = Levy(x_j^t) \tag{2.54}$$

Then α is a weighting factor which is calculated as follows:

$$\alpha = S_{\max} / t^2 \tag{2.55}$$

where;

$S_{\max}$ is the maximum walk step that a monarch butterfly individual can move in one step and,

t is the current generation time.

The bigger the value of α, signifying long step of search, increases the influence of dx on $x_{j,k}^{t+1}$ and enhance the exploration process. Smaller value of α, indicates short step of search, decreases the influence of dx on $x_{j,k}^{t+1}$ and enhance exploitation. The pseudo code implementation of the MBO adjusting operator is given as follows (Gai-Ge, *et al.* 2016; Gai-Ge, *et al.* 2015):

Algorithm 2: Butterfly adjusting operator

Begin

for *j=1* to NP_2 (for all monarch butterflies in Subpopulation 2) **do**
calculate the walk step *dx*;
calculate the weighting factor;
for *k=1* to D (all the elements in the *jth* monarch butterfly) **do**
Randomly generate a number *rand* by uniform distribution;
if $r \leq p$ **then**
Generate the *kth* element of x_j^{t+1}
else
Randomly select a monarch butterfly in Subpopulation 2 (say r_3);
Generate the *kth* element of x_j^{t+1}
if $r > BAR$ **then**
$x_{j,k}^{t+1} + \omega \times (\mathrm{d}x_k - 0.5)$
end if
end if
end for *k*
end for *j*

End.

The schematic pseudo code representation of the MBO algorithm, showing the parameter initialization, population generation and fitness evaluation is given as follows (Gai-Ge, *et al*. 2016; Gai-Ge, *et al.* 2015):

Algorithm 3: Monarch Butterfly Optimization Algorithm
Begin
Step 1: Initialization. Set the generation counter *i=1*; initialize population P of *NP* monarch butterfly individuals randomly; set the maximum generation *MaxGen*, monarch butterfly number NP_1 in Land 1 and monarch butterfly NP_2 in Land 2, max step S_{max}, butterfly adjusting rate *BAR*, migration period *peri*, and the migration ratio *p*
Step 2: Fitness evaluation. Evaluate each monarch butterfly according to its position
Step 3: While the best solution is not found **or** $t < MaxGen$ **do**
Sort all the monarch butterfly individuals according to their fitness
Divide monarch butterfly individuals into two subpopulations (Land 1 and Land 2)
for *i=1* to NP_1 (for all monarch butterflies in Subpopulation 1) **do**
Generate new Subpopulation 1 according to Algorithm 1
end for *i*
for *j=1* to NP_2 (for all monarch butterflies in Subpopulation 2) **do**
Generate new Subpopulation 2 according to Algorithm 2
end for *j*
Combine two newly-generated subpopulations into one whole population;
Evaluate the population according to the newly updated populations;
t=t+1
Step 4: end while
Step 5: Output the best solution
End.

2.2.7 IEEE standard test system

These are standard test benchmarks standardized by the Institute of Electrical and Electronics Engineers in order to give researchers a common ground for research, results testing and comparison. Standard ELD test benchmarks include: The IEEE 3-unit generators system, the IEEE 5-unit generators system 14-bus system, the 30-bus IEEE test network with 6-unit generators, the IEEE 15-unit generators systems, the IEEE 40 unit test system etc. (Kassim and Wafaa 2012). For the purpose of this research, the proposed method was implemented on IEEE standard 3-units ELD test system, IEEE 6-units ELD test system and a 15-units ELD test system. Detail information on the IEEE standard ELD test system can be found in (Dasgupta, *et al.* 2016; Sinha, *et al.* 2003).

2.2.7.1 The IEEE 3-Unit Generators System

The 3-unit test system is one of the simplest ELD test systems which is modeled with a valve point effect cost of generation. The details characteristics of 3-unit test system such as coefficient of the fuel cost, the maximum and minimum power limits are given in Table 3.1. The total power demand of the system is 150 MW (Sinha, *et al.* 2003).

Table 2.1: IEEE 3-unit ELD Test System Data

Generator	$P_{\min}$	$P_{\max}$	a	b	c	e	f
1	150	600	561	7.92	0.001562	300	0.0315
2	100	400	310	7.85	0.001940	200	0.0420
3	50	200	78	7.97	0.004820	150	0.0630

The B-loss coefficients of the system are given as:

$$B_{ij} = \begin{bmatrix} 0.00670 & 0.00953 & -0.00507 \\ 0.00953 & 0.05210 & 0.00901 \\ -0.00507 & 0.00901 & 0.00294 \end{bmatrix} \tag{2.56}$$

$$B_{oi} = \begin{bmatrix} -0.7660 & -0.0342 & 0.1890 \end{bmatrix} \times 10^{-4} \tag{2.57}$$

$$B_{oo} = 4.0357 \times 10^{-2} \tag{2.58}$$

2.2.7.2 The IEEE 6-Unit Generators System

The 6-unit test system is another IEEE test system which has six (6) thermal units, twenty-six (26) buses and forty-six (46) transmission lines. It is with a ramp-rate limit as constraints and has a base capacity of 100MVA. The detailed characteristics of 6-unit test system such as coefficient of the fuel cost, the maximum and minimum power limits are given in Table 2.2.

Table 2.2: IEEE the 6-units ELD Test System Data

Generator	P_{min}	P_{max}	a	b	c	P^{now}	R^{up}	R^{down}
1	100	500	240	7.0	0.0070	440	80	120
2	50	200	200	10.0	0.0095	170	50	90
3	80	300	220	8.5	0.0090	200	65	100
4	50	150	200	11.0	0.0090	150	50	90
5	50	200	220	10.5	0.0080	190	50	90
6	50	120	190	12.0	0.0075	110	50	90

The B-loss coefficients of the system are given as:

$$B_{ij} = \begin{bmatrix} 0.0017 & 0.0012 & 0.0007 & -0.0001 & -0.0005 & -0.0002 \\ 0.0012 & 0.0014 & 0.0009 & 0.0001 & -0.0006 & -0.0001 \\ 0.0007 & 0.0009 & 0.0031 & 0.0000 & -00010 & -0.0006 \\ -0.0001 & 0.0001 & 0.0000 & 0.0024 & -0.0006 & -0.0008 \\ -0.0005 & -0.0006 & -0.0010 & 0.0129 & 0.0129 & -0.0002 \\ -0.0002 & -0.0001 & -0.0006 & -0.0008 & -0.0002 & 0.0150 \end{bmatrix} \quad (2.59)$$

$$B_{oi} = \begin{bmatrix} -0.3908 & -0.1297 & 0.7047 & 0.0591 & 0.2161 & -0.6635 \end{bmatrix} \times 10^{-3} \quad (2.60)$$

$$B_{oo} = 0.056 \quad (2.61)$$

2.2.7.3 The IEEE 15-Unit Generators System

The IEEE 15-units ELD test system is one of the most widely used economic load dispatch test systems used for validating the performance of a developed ELD solution methods. It contains Fifteen (15) generating units and can be modeled with ramp-rate limits and prohibited operating zones as constraints. The system has a power demand of two thousand six hundred and thirty-megawatt (2630MW) power demand with a capacity and its data is given in Table 2.3 as follows:

Table 2.3: IEEE 15-units ELD Test System Data

Generator	P_{min}	P_{max}	a	b	C
1	150	445	671	10.1	0.000299
2	150	445	574	10.2	0.000183
3	20	130	374	8.8	0.001126
4	20	130	374	8.8	0.001126
5	150	470	461	10.4	0.000205
6	135	460	630	10.1	0.000301
7	60	465	548	9.8	0.000364
8	20	300	227	11.2	0.000338
9	25	162	173	11.2	0.000807
10	25	160	175	10.7	0.001203
11	20	80	186	10.2	0.003586
12	20	80	230	9.9	0.005513
13	25	85	225	13.1	0.000371
14	15	55	309	12.1	0.001929
15	15	55	323	12.4	0.004447

The ramp rate limits and the prohibited operating zones for the IEEE Units data for the 15-units ELD test system are given in Table 2.4.

Table 2.4: IEEE 15-Units Data with Ramp Rate Limits and Prohibited Operating Zones

Generator	P^0(MW)	UR(MW/h)	DR(MW/h)	Prohibited zones
1	400	80	120	--
2	300	80	120	[185 225]; [305 335]; [420 450]
3	105	130	130	--
4	100	130	130	--
5	90	80	120	[180 200]; [305 335]; [390 420]
6	400	80	120	[230 255]; [365 395]; [430 455]
7	350	80	120	--
8	95	65	100	--
9	105	60	100	--
10	110	60	100	--
11	60	80	80	--
12	40	80	80	[30 40]; [55 65]
13	30	80	80	--
14	20	55	55	--
15	20	55	55	--

The B-loss coefficients of the system are given as:

$$B_{ij} = \begin{bmatrix} 1.4 & 1.2 & 0.7 & -0.1 & -0.3. & -0.1 & -0.1 & -0.1 & -0.3 & 0.5 & -0.3 & -0.2 & 0.4 & 0.3 & -0.1 \\ 1.2 & 1.5 & 1.3 & 0 & -0.5 & -0.2 & 0 & 0.1 & -0.2 & -0.4 & -0.4 & 0 & 0.4 & 1.0 & 0.2 \\ 0.7 & 1.3 & 7.6 & -0.1 & -1.3 & -0.9 & -0.1 & 0 & -0.8 & -1.2 & -1.7 & 0 & -2.6 & 11.1 & -2.8 \\ -0.1 & 0 & -0.1 & 3.4 & -0.7 & -0.4 & 1.1 & 5.0 & 2.9 & 3.2 & -1.1 & 0 & 0.1 & 0.1 & -2.6 \\ -0.3 & -0.5 & -1.3 & -0.7 & 9.0 & 1.4 & -0.3 & -1.2 & -1.0 & -1.3 & 0.7 & -0.2 & -0.2 & -2.4 & -0.3 \\ -0.1 & -0.2 & -0.9 & -0.4 & 1.4 & 1.6 & 0 & -0.6 & -0.5 & -0.8 & 1.1 & -0.1 & -0.2 & -1.7 & 0.3 \\ -0.1 & 0 & -0.1 & 1.1 & -0.3 & 0 & 1.5 & 1.7 & 1.5 & 0.9 & -0.5 & 0.7 & 0 & -0.2 & -0.8 \\ -0.1 & 0.1 & 0 & 5.0 & -1.2 & -0.6 & 1.7 & 16.8 & 8.2 & 7.9 & -2.3 & -3.6 & 0.1 & 0.5 & -7.8 \\ -0.3 & -0.2 & -0.8 & 2.9 & -1.0 & -0.5 & 1.5 & 8.2 & 12.9 & 11.6 & -2.1 & -2.5 & 0.7 & -1.2 & -7.2 \\ -0.55 & -0.4 & -1.2 & 3.2 & -1.3 & -0.8 & 0.9 & 7.9 & 11.6 & 20.0 & -2.7 & -3.4 & 0.9 & -1.1 & -8.8 \\ -0.3 & -0.4 & -1.7 & -1.1 & 0.7 & 1.1 & -0.5 & -2.3 & -2.1 & -2.7 & 14.0 & 0.1 & 0.4 & -3.8 & 16.8 \\ -0.2 & 0 & 0 & 0 & -0.2 & -0.1 & 0.7 & -3.6 & -2.5 & -3.4 & 0.1 & 5.4 & -0.1 & -0.4 & 2.8 \\ 0.4 & 0.4 & -2.6 & 0.1 & -0.2 & -0.2 & 0 & 0.1 & 0.7 & 0.9 & 0.4 & -0.1 & 10.3 & -10.1 & 2.8 \\ 0.3 & 1.0 & 11.1 & 0.1 & -2.4 & -1.7 & -0.2 & 0.5 & -1.2 & -1.1 & -3.8 & -0.4 & -10.1 & 57.8 & -9.4 \\ -0.1 & -0.2 & -2.8 & -2.6 & -0.3 & 0.3 & -0.8 & -7.8 & -7.2 & -8.8 & 16.8 & 2.8 & 2.8 & -9.4 & 128.3 \end{bmatrix} \times 10^{-3} \quad (2.62)$$

$$B_{oi} = \begin{bmatrix} -0.1 & -0.2 & 2.8 & -0.1 & 0.1 & -0.3 & -0.2 & -0.2 & 0.6 & 3.9 & -1.7 & 0 & -3.2 & 6.7 & -6.4 \end{bmatrix} \times 10^{-3} \quad (2.63)$$

$$B_{oo} = 0.055 \quad (2.64)$$

2.3 Review of Similar Works

Some of the related literatures which are relevant for providing basic empirical work carried out by different researchers for the successful implementation of the proposed research work are discussed as follows.

Kumar and Suchitra (2011) developed a bi-objective Economic Load Dispatch (ELD) problem considering a wind turbine. A fuzzy membership function was developed to represent the dispatch of wind power into the conventional system. Particle swarm optimization algorithm, Genetic algorithm and bacteria-foraging optimization algorithm technique were adopted to develop an economic load dispatch scheme compromising both the economic and security requirements. The results of all these 3 proposed techniques were compared. Numerical analyses were reported based on a typical IEEE-30-bus with six-generator test power system to show the validity and applicability of the proposed approaches. Results showed that all the proposed methods were efficient with the particle swarm optimization having a strong convergence time and the ant colony optimization having a better precision. However, this work did not consider the influence of environmental constrains, ramp rate effect and valve loading in modeling the ELD problem. Thus, the method is limited in terms of depicting the real-life situation of the ELD problem.

Rani and Kothari (2012) presented an efficient solution to dynamic economic emission load dispatch (DEED) problem. In this paper, a Chaotic Self Adaptive Particle Swarm Optimization (CSAPSO) algorithm was developed to solve the DEED problem. The work considered some non-linear characteristics of the generator such as, valve point effect, non-smooth cost functions of the fuel and emission and ramp rate limits. The valve point effects were modeled and imposed as rectified sinusoid components. The velocity of the PSO was modeled

dynamically in order to increase its precision. A chaotic local search operator was introduced in the algorithm in order to avoid premature convergence. Simulation was carried out in MATLAB environment where fuel cost and emission are treated as competing objectives. The applicability and high feasibility of the method were validated on 10-unit test systems and results demonstrated the efficiency of the proposed method over the standard particle swarm optimization. However, this work only considered the non-smooth cost function constraint and valve loading effecting which are enough to depict and provide an optimization solution to the practical ELD problem.

Ghasemi (2012b) proposed a multi objective particle swarm optimization (MOPSO) for Dynamic Economic Load Dispatch (DELD) problem solution with transmission losses. The proposed algorithm is based on multi objective meta-heuristics technique that evaluates a set of the pareto solutions systematically and preserve the diversity measure tactic. Simulation result performed on the 6 and 15-unit test systems considering various demands for 24 hours showed the effectiveness of the proposed MOPSO approach to generate well distributed pareto optimal non-dominated solution of multi-objective DELD problems. However, the work only considered valve point effect which is not enough in providing an optimal solution to the ELD problem because other constraints such as ramp rate and multi-fuel effects makes the ELD solution more practicable.

Sayah and Hamouda (2013) employed the use of Differential Evolution (DE) and Particle Swarm Optimization (PSO) as hybrid technique due to their advantages and disadvantages. They proposed the use of hybrid Differential Evolution and Particle Swarm Optimization called Differential Evolution Particle Swarm Optimization (DEPSO). The method incorporated the PSO procedure as a supplementary mutation operator into the conventional DE algorithm to improve the global search capability, and to prevent premature convergence to local minima. The proposed DEPSO consist in a strong cooperation of the two evolutionary algorithms due to

the fact that DE advantages to maintain the diversity of population and to explore local search while, PSO has memory capability. The proposed method considered prohibited operating zones, ramp rate and losses but failed to consider valve point effects, multiple fuel effects and spinning reserve as constraints which in real economic load dispatch affects system operation and cost of generation**.**

Singh and Kumar (2013) focused on minimizing the total generation cost of thermal units as well as to minimizing the pollutant emission by toxic gases. The research work developed a modified particle swarm optimization with a moderate random search strategy called the moderate-random particle swarm optimization (MRPSO). This is with the aim of enhancing the ability of the PSO to explore the solution space more effectively and increase its convergence rates. The developed MRPSO was used to solve the economic load dispatch problem considering the emission constraint. The validation of the proposed MRPSO algorithm was demonstrated through its application for six generator systems with emission constraints for various load demands. Results obtained showed the capability of the proposed method in minimizing the cost of generation but the research did not incorporate the ramp rate constraint and the unit commitments which decide whether a generating unit is either committed or de-committed over a planning horizon with the ELD problem. Thus, the method proposed in this research work may not be efficient in practical ELD problem.

Tiwari *et al.*, (2013) proposed a classical particle swarm optimization technique to solve constraints based economic load dispatch problem with generator constraints and power losses. The algorithm was tested on three (3) and six (6) generator units. Results were compared with Genetic algorithm and lambda iteration method. The method proposed in this work showed an efficient method for solving the economic load dispatch (ELD) for different size of power system. However, the method considered only two constraints, generator constraints and power

loss. Also, other constraints such as valve point effect, ramp rate, prohibited operating zones which represent a near practical scenario in real world application were not considered.

Elyas, *et al.* (2014) present an efficient approach for solving the economic load dispatch (ELD) problem with valve-point effect using a new hybrid optimization algorithm. The hybrid algorithm is based on Clonal Selection Algorithm which employed the feature of Gaseous Brownian Motion Optimization (GBMO) and Particle Swarm Optimization (PSO) for local search and improving and ensuring an efficient generation of the initial position respectively. The performance of the developed method was tested on two economic load dispatch test system considering a set of constraints. Simulation results were compared with similar method available in literature. Simulation results demonstrated the effectiveness of the proposed method on efficiently addressing the problem of economic load dispatch. However, this work did consider some of the ELD constrains such as valve loading effect. Also, hybridizing the CSA, the GMBO and the PSO increases the convergence capability of the hybridized algorithm. Thus, may take a longer time before the optimized ELD solution is obtained.

Subathra, *et al.* (2015) Presented a new hybrid approach integrating the cross-entropy (CE) algorithm and the sequential quadratic programming (SQP) technique to solve the Economic Load Dispatch (ELD) problem related to electrical power generating units. Due to the introduction of the valve-point effect in the ELD objective function, the optimization task required tools appropriate for a non-convex optimization landscape**.** In this respect, the CE approach was employed which constructs a random sequence of solutions probabilistically converging to a near-optimal solution and, thus, facilitating the exploration capability. Additionally, to fine-tune the solution in promising basins of attraction, the SQP algorithm is invoked, which performs a local search. Despite its reliance on a global heuristic scheme, CE-SQP is vested with fast convergence capability, which may entail its use for online power dispatch. However, this work failed to integrate some additional constraints such as ramp rate,

environmental constraint and the unit commitment problem that decides whether a generating unit is either committed or de-committed over a planning horizon with the ELD problem.

Dasgupta, *et al.* (2015) proposed a solution to economic load dispatch problem using four (4) different types of particle swarm optimization algorithm. The algorithms include the standard Particle Swarm Optimization technique (PSO), Particle Swarm Optimization with Constriction Factor Approach (PSOCFA), Particle Swarm Optimization technique with Inertia Weight Factor Approach (PSOIWA) and Particle Swarm Optimization technique with Constriction Factor and Inertia Weight Factor Approach (PSOCFIWA). The economic load dispatch prohibited zone constraint has been considered. Simulation results and comparative analysis of fuel cost through all the four optimization techniques were discussed. Results demonstrated that the proposed methods are efficient in addressing the economic load dispatch problem. However, this work only considered the prohibited operating zone of the ELD problem. This is not efficient to depict the practical scenario of the ELD problem. Thus, the method may not be valid for all ELD problems.

Bahrani and Patra (2015) Proposed an enhanced Particle Swarm Optimization (PSO) algorithm named Orthogonal PSO (OPSO) algorithm, for Economic Load Dispatch (ELD) of the generated power in a smart grid environment. The equality and inequality constraints and power balance response against mismatch between load demand and total power outputs of generating units were considered. In the OPSO algorithm an Orthogonal Vectors (OVs) is applied in the d-dimensional search space. The particles that have possible solutions were made to move in the d-dimensional search space to form OVs. The performance of the proposed technique was tested and evaluated using 15 generating units. Simulation results were compared with similar algorithm and the standard PSO and the results shows that the OPSO algorithm provided a better solution in terms of precision and convergence capability.

However, this work also failed to integrate the unit commitment problem that decides whether a generating unit is either committed or de-committed over a planning horizon with the ELD problem

Trivedi, *et al.* (2016b) Presented the use of a meta-heuristic algorithm called JAYA optimization algorithm was implemented for solving of the economic load dispatch problem in the MATLAB simulation environment. The minimization of total cost was obtained for two different scenarios such as all sources except solar energy and all sources except wind energy. In both scenarios, the result shows the comparison of JAYA Algorithm with the reduced gradient method for the economic load dispatch problem solution. The results obtained with JAYA Algorithm gave a better cost reduction in less iteration as compared to reduced gradient method which showed the effectiveness of JAYA algorithm in solving the ELD problem. However, evaluating the performance of JAYA only with reduced gradient method may not be sufficient for a concise conclusion. Also, the number of constraints considered in the work was not sufficient to depict the practical situation of ELD problem.

Santra, *et al.* (2016) presented a novel solution of convex and non-convex economic load dispatch (ELD) problem of small scale thermal power system using a hybrid Particle Swarm Optimization (PSO) and Ant Colony Optimization (ACO) algorithms. The particle swarm optimization was used to perform the ELD cost minimization while the solution obtained was optimized (tuned) by the ant colony optimization. The technique was applied to a three (3) generator five (5) bus test system by considering generator capacity constraints, transmission loss, ramp rate limits, prohibited operating zones and valve point loading. Simulation results showed that the proposed method is efficient with an improved performance. However, this work did not consider the influence of environmental constraint, ramp rate effect and valve

loading in modeling the ELD problem. Thus, the method is limited in terms of depicting the real-life situation of the ELD problem.

From the literatures reviewed, it is evident that the economic load dispatch problem which is a nonlinear multi-constraints optimization problem has been given a significant research attention. Several efforts have been made to address these problems using the standard methods and significant research effort have also been using computational intelligence algorithm. However, most researchers did not pay attention into considering the influence of practical constraints and the unit commitment at the same time which makes the solutions quite unrealistic as optimal solutions are not guaranteed when some important constraints are not considered. Thus, this research has developed an efficient method for solving the ELD problem considering the practical constraints using the monarch butterfly optimization.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

This chapter is divided into two subsections. In the first subsections, the materials which were used for the successful implementation of the research are presented. In the second subsection, all the methods and mathematical models employed in developing the proposed monarch butterfly optimization based economic load dispatch are presented.

3.2 Materials

In this section, the materials used for the implementation of the research presented in this report are discussed. These materials involve the specification of the computer system used and the software used for implementation.

3.2.1 MATLAB Simulation Environment

All the models implemented in this research were simulated in MATrix LABoratory (MATLAB) software. The Monarch butterfly optimization was programmed into a MATLAB file (mfile) called MBO.m using the programming window (editor) of the MATLAB software. The economic load dispatch problem was programmed into an object function called "eldCost.m". The IEEE test system was imported into the "eldCost.m" program from excel file using the "xlsread" command and the "MBO.m" was used to evaluate the optimum of the "eldCost.m" function. Figure 3.1 shows the programming interface of MATLAB software.

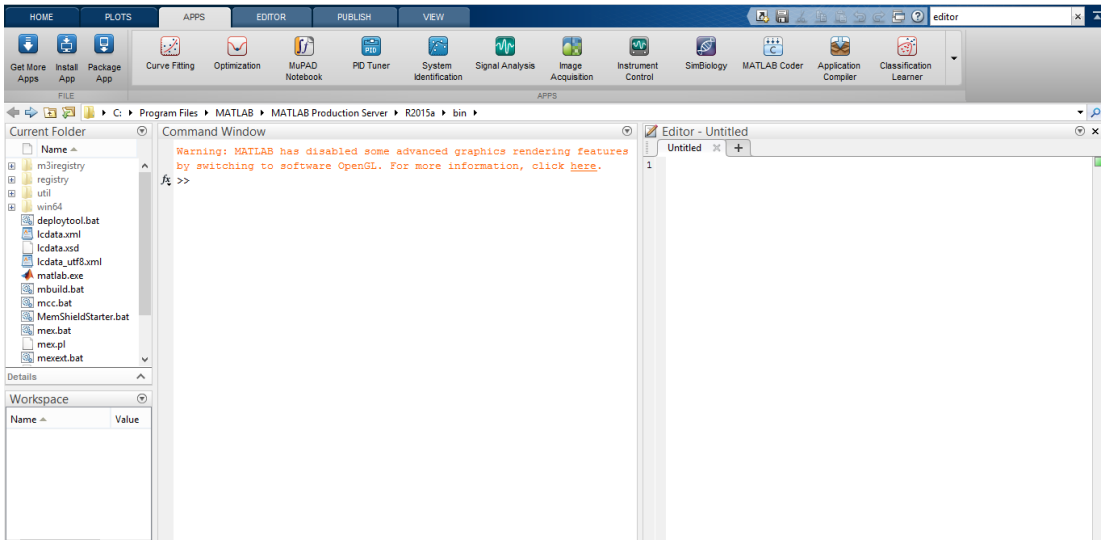


Figure 3.1: MATLAB Simulation Environment

3.2.2 IEEE test system

The IEEE test is standardized network data sets available for testing the performance of any developed model for economic load dispatch. There are various scenarios of data set available for evaluating the performance of developed methods, but there is no standard one that has to be followed in selecting any test systems. Thus, for the purpose of this research and for the purpose of validation, three different standard test systems were used. These are 3-units, 6-units and 15-units IEEE economic load dispatch test systems. These test systems are discussed as follows:

I. IEEE 3-Unit Test System

The 3-unit test system is one of the simplest ELD test systems for economic load dispatch testing and validation. It is modeled with a valve point effect cost of generation as described in subsection 2.2.7.1. The loss coefficients used to model this test system on MATLAB are given in equations (2.56), (2.57), and (2.58) respectively. The characteristic data of the system showing the minimum and maximum power of each generating unit was also presented in Table 2.1.

II. IEEE 6-unit test system

The 6-unit test system was also used in this work to check the viability of the algorithm for ELD problems including ramp rate as constraint. The system is explicitly described in subsection 2.2.7.2 while the system data including ramp-rate limit of each generator used in modelling the system on MATLAB is given in Table 2.2. Also, the loss coefficients are shown in equations (2.59), (2.60), and equation (2.61) respectively.

III. IEEE 15-unit test system

The IEEE 15-units ELD was the last test system used in this work. It is a very robust system with more practical applicability as fully described in subsection 2.2.7.3. The data of the system was provided in Table 2.3 and Table 2.4 respectively. However, Table 2.4 only shows the data of ramp rate limits and prohibited operating zones in the system which is one of the reasons why the 15-unit system is considered to be robust. The loss coefficients used together with the system data to model the system on MATLAB is given in equations (2.62), (2.63) and (2.64) respectively.

The three different test systems used in this research work were considered so as to ensure that the developed approach has maximum practical applicability to a real scenario of economic load dispatch in multiple generators connected in a network. This is to ensure that approach would be adequate for any kind of load dispatch problem.

3.3 Methods

In this section, the step by step procedure employed in the development of the monarch butterfly optimization based economic load dispatch is discussed. The metrics used to evaluate the performance of the developed method are also presented.

3.3.1 Monarch Butterfly Optimization (MBO) for ELD problem

In this dissertation, the MBO was used as optimization tool for solving the constrained ELD problem. At the initial stage the MBO was replicated and simulated in MATLAB environment this is achieved using the following steps

Step 1: In the first stage, all the parameters of the MBO and the ELD cost functions were initialized. The MATLAB code for parameter initialization of the MBO is presented in Appendix B. These parameters include the population of the monarch butterflies, migration period, monarch butterfly ratio, butterfly adjustment rate, movement step of butterflies and number of iteration. And the ELD parameters initialized includes the number of decision variables (depending on the test system evaluated), The B-coefficients, capacities and cost coefficients and the power demand. These parameters were used as modeling parameters for the development of the method developed in this dissertation.

Step 2: In this step, the positions of the monarch butterflies were randomly initialized using equation (3.1) as follows:

$$NP = rand(N, D) \tag{3.1}$$

where;

N, is the total number of butterflies and

D, is the number of decision variables which is function of the ELD test system the algorithm is solving.

The fitness of the randomized position of the butterflies was evaluated and the butterflies which had the best position was determined and stored in the memory accordingly.

Step 3: In this step, the maximum generation counter is initialized and the individual generation butterflies are sorted according to their fitness in step two. The population of the MBO was sorted using the code presented in Appendix C thus dividing them into subpopulation one (land 1-P_1) and subpopulation two (land 2-P_2) populations using equations (3.2) and (3.3) respectively.

$$P_1 = \lceil p \times NP \rceil \tag{3.2}$$

$$P_2 = NP - P_1 \tag{3.3}$$

where;

[p x NP] is an indication that the population at land one is rounded up to the nearest positive integer greater or equal to the total number of butterfly's population (*NP*) and

p is the ratio of butterflies in land 1.

The migration process of the monarch butterflies was then expressed using the equation described in (2.55). If the migrated position of the butterflies is less or equal to the butterflies' ration in Land 1, then a new position is generated using equation (3.4)

$$r = rand \times peri \tag{3.4}$$

Where, *peri* indicates the migration period of the butterflies.

Step 4: A loop was initiated for the monarch butterflies in subpopulation one (Land 1) and the new subpopulation was generated using equation (2.53). Another loop was initiated for the monarch butterflies in subpopulation two (Land 2) and a new subpopulation was generated using equation (2.54).

Step 5: The two newly generated populations in step four were combined and the fitness of the new positions were evaluated. The evaluated fitness was ranked and the fitness obtained by the butterfly with the best position is outputted as the result. This process was repeated until the entire iteration process is completed.

These steps highlighted were used to implement the MBO for ELD optimization while the flowchart of this procedure is presented in Figure 3.2. The objective function of the ELD was programmed into an objective function whose fitness is evaluated using the MBO. The objective function model of the ELD is discussed in subsection 3.3.2

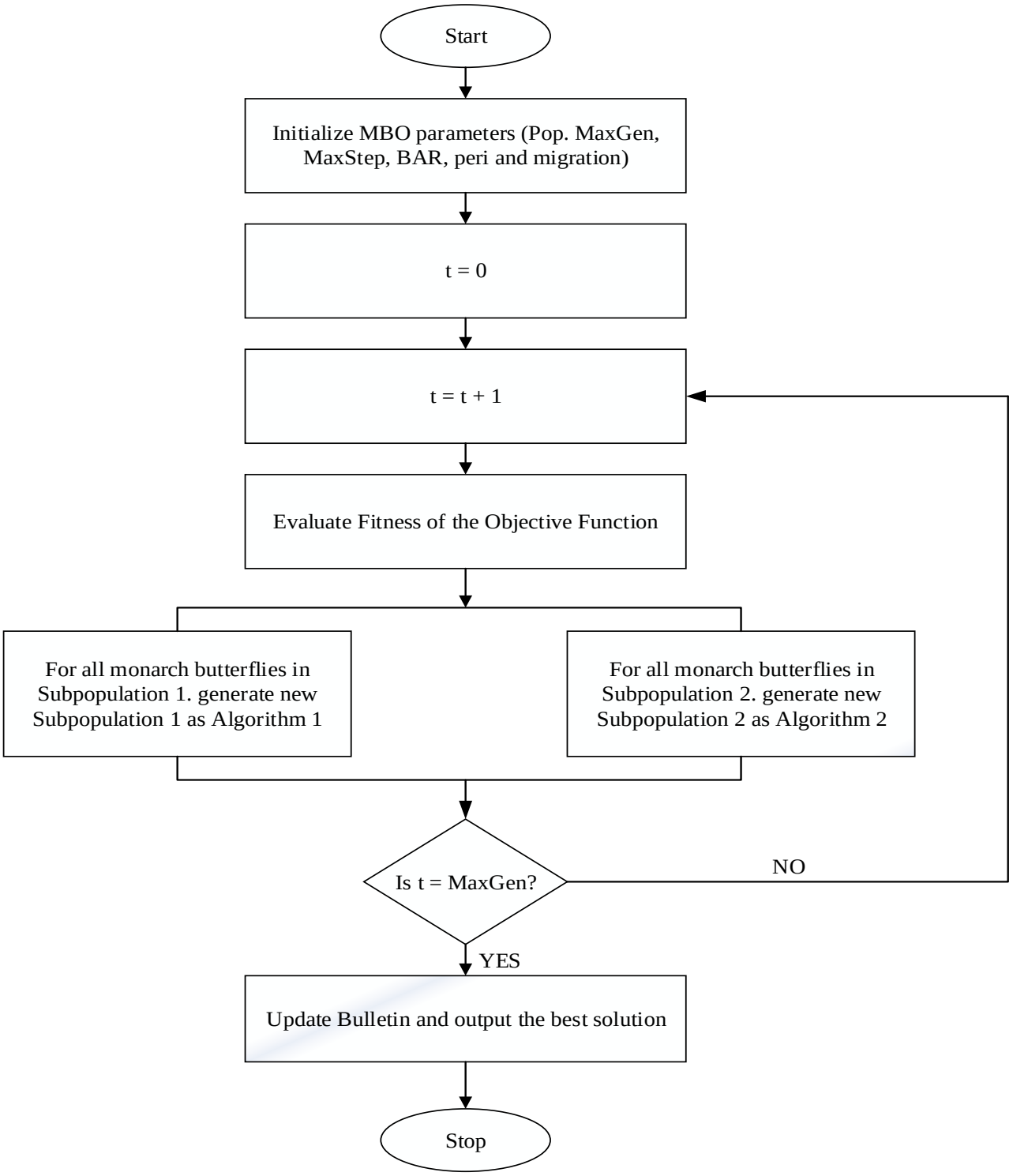


Figure 3.2: Flowchart of MBO (Wang *et al.*, 2015)

From Figure 3.2, the process of MBO was initialized and the MBO parameters, total number of iteration and the initial population of the monarch butterfly were provided. The fitness of the initial population was evaluated and the butterfly population was divided into two

subpopulations (Land1 and Land2). For all monarch butterflies in Land1, a new population was generated using the migration operator described in subsection 2.2.6.1. For all the monarch butterfly in Land2, a new population was also generated using the adjustment operator described in subsection 2.2.6.2. The process was repeated until the total number of iteration was reached. The butterfly with the best position is stored and outputted as the optimum result. The Parameters of the MBO used for simulation are given in Table 3.4

Table 3.1: Parameters of the MBO Used for Simulation

SN	Parameter	Symbol	Value
1	Total Population	NP	50
2	Population	NP1	21
3	Population	NP2	29
4	Period	peri	1.2
5	Adjustment Rate	BAR	0.4167
6	Dimension	D	3,6,15

3.3.2 Economic load dispatch problem modeling

Recall that economic load dispatch involve determining the optimum real power settings of power system generating units with an objective of minimizing the total operating cost over a dispatch period while satisfying a number of constrains. To achieve this, the objective function (whose MATLAB script is shown in Appendix D) focused on minimizing the overall cost of generating units for the load profiles given in subsection 3.2.2. The objective function is given in equation (3.5).

$$C = \min \sum_{t=1}^{T} \sum_{i=1}^{N} F_i\left(P_i^t\right) \tag{3.5}$$

where;

N is the number of generator,

T is the number of runs,

F_i is the fuel cost for the generating unit (\$/h) i which is expressed in equation (3.6) as.

$$F_i\left(P_i^t\right) = a_i + b_i P_i^t + c_i {P_i^t}^2 \tag{3.6}$$

where,

P_i^t is power output of generator i in time interval t;

a_i, b_i and c_i are cost coefficients function of generating unit i, respectively.

The developed MBO in this research was used to evaluate the fitness of the cost function given in equation (3.5), considering three (3) different constraints as follows:

A. Transmission Loss Constraints

The condition for this constraint which is also called the equality constraints for a given time interval t is given as:

$$\sum_{i=1}^{N} P_i^t - P_D^t - P_{loss}^t = 0 \tag{3.7}$$

where;

P_i^t is the power injected at bus i,

P_D^t is the total power demand,

P_{loss}^t, is the losses which depend on the generated power of any active unit and changes by new generation.

Since the transmission loss is expressed as a function of generator power output through B-coefficients, under normal operating conditions, the transmission loss is quadratic in the injected bus real power and is expressed as:

$$P_{loss}^{t} = \sum_{i=1}^{N}\sum_{j=1}^{N} P_i^t B_{i,j} P_j^t + \sum_{i=1}^{N} B_{i,0} P_i^t + B_{00} \tag{3.8}$$

where;

P_i^t and P_j^t are injected real power at *ith* and *jth* buses,

$B_{i,j}$ is the loss coefficient,

$B_{i,0}$ and B_{00} are constants matrices for transmission loss.

B. Generation Capacity Constraints

In order to ensure an efficient operation, the generator outputs are constrained by some upper and lower limits constraints given in equation (3.9)

$$P_i^{\min} \le P_i^t \le P_i^{\max} \tag{3.9}$$

where;

$P_i^{\min}$ and $P_i^{\max}$ are minimum and maximum active power of the *ith* generator respectively.

C. Ramp Rate Limit Constraints

This constrain limit the range of actual online generating units. Thus, the generator constrains due to ramp rate limits of *ith* generating unit is given in equations (3.10) and (3.11).

$$P_i \le P_i^0 \le UR_i \tag{3.10}$$

$$P_i^0 \le P_i \le DR_i \tag{3.11}$$

The capability of *ith* generating units with ramp rate consideration is given in equation (3.12) as:

$$\max(P_i^{\min}, P_i^0 - DR_i) \le P_i \le \min\left(P_i^{\max}, P_i^0 + UR_i\right) \tag{3.12}$$

Thus, to ensure efficient an efficient fitness evaluation in this research, the maximum and minimum output power limits equations (3.13) and (3.14) respectively.

$$P_i^{\max-ramp} = \min\left(P_i^{\max}, P_i^{t+1} + UR_i\right) \tag{3.13}$$

$$P_i^{\min-ramp} = \min\left(P_i^{\min}, P_i^{t-1} + DR_i\right) \tag{3.14}$$

where: UR and DR refer to the up-ramp rate and down ramp rate respectively.

The economic load dispatch model described in this subsection (i.e. 3.3.2) was programmed into an objective function file called *ELDcost.m* which the MBO used to evaluate the fitness. The flow chart for implementing the programming of the proposed MBO based algorithm for ELD considering unit commitment is given in Figure 3.3.

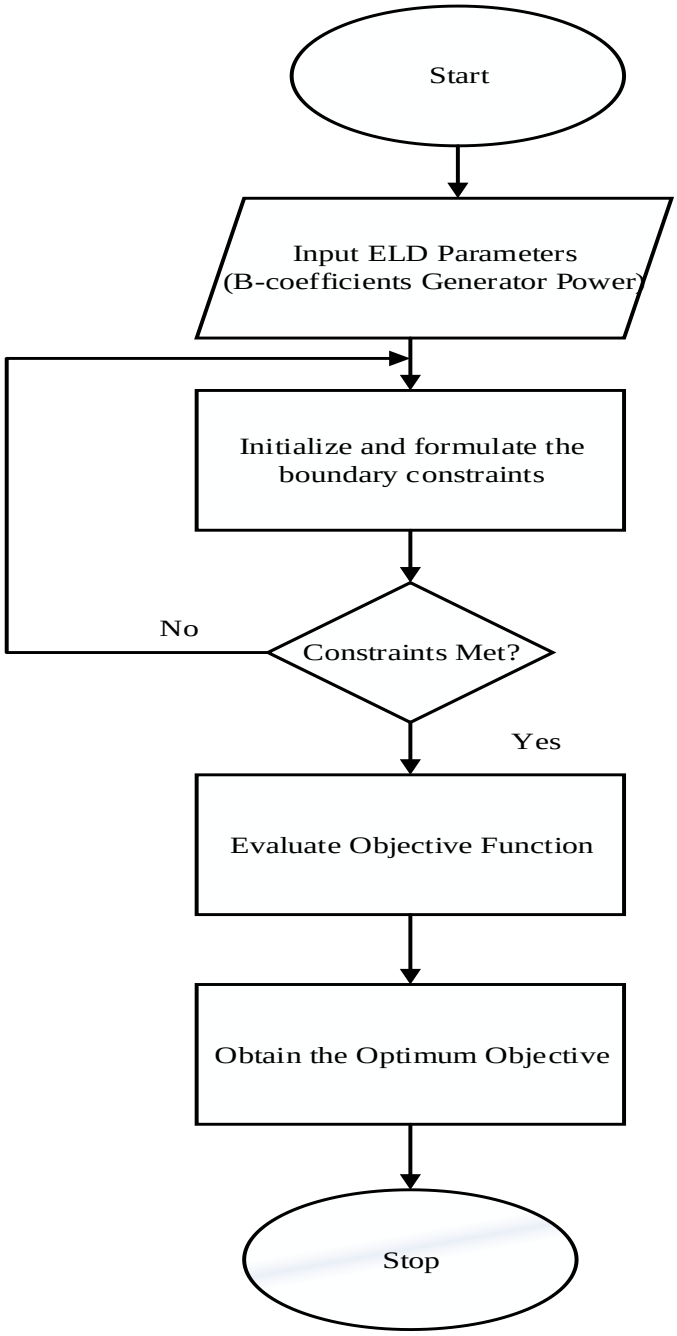


Figure 3.3: Flow Chart for Implementation of ELD Cost Function

Figure 3.3 shows the implementation flow chart of Economic Load Dispatch (ELD) cost functions. The process of the cost evaluation begins by providing the ELD data and the loss coefficient. The generation, prohibited operating zones constraints were initialized and the generation cost given in equation (3.4) was evaluated using the MBO algorithm described in subsection 3.3.1.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

This chapter present the result of the monarch butterfly optimization based economic load dispatched model developed in this dissertation. The developed model was tested on standard IEEE 3-uints ELD test system, IEEE 6-units ELD test system and IEEE 15-units ELD test system and results obtained on each system were presented and discussed. Validation using, the work presented in (Sayah and Hamouda, 2013) are also discussed.

4.2 Simulation Result of the Proposed Model on IEEE 3-Units ELD Test System

The parameters of the monarch butterfly optimization algorithm given in subsection 3.3.1 of chapter three were employed alongside the data of the 3-unit system given in subsection 2.2.7.1 in chapter two for the simulation. The MBO was used to evaluate the economic load dispatch cost function and the subplots showing the minimization process of MBO considering a power demand of 150 megawatts and 300 megawatts is given in Figure 4.1.

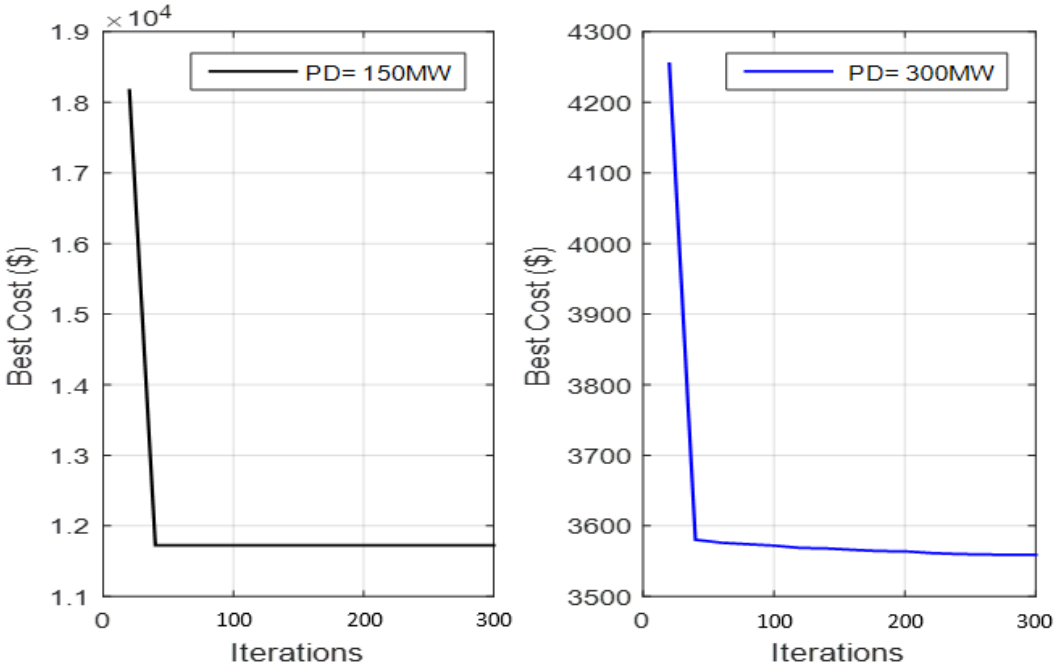


Figure 4.1: Cost Minimization Plot on 3-Unit System

From Figure 4.1, it can be observed that, the monarch butterfly optimization algorithm obtained an optimized result for the ELD 3-unit IEEE test system. During the simulation, the optimization process was evaluated over a total of 300 iterations to ensure that the MBO converges towards the optimum results before the end of the iteration process. On the 3-unit test system, it can be observed that the MBO was able to minimize the cost of generation efficiently as can be seen on the Best Cost axis of Figure 4.1. Comparing the 150 MW and 300 MW, it can be observed that the MBO obtained some converged results of $11,722.4130$ at about 50 iterations and $3,561.3973 at about 50 iterations respectively. Thereafter, the results remain the same all through the iteration process. Details of the results obtained for the 3-uint system is given in Table 4.1.

Table 4.1: Best Solution Obtained by Proposed MBO Based ELD Algorithm for IEEE 3-Unit ELD Test System

Generator	**Power (150 MW)**	**Power (300 MW)**
P1	56.6114	150.7313

P2	54.4011	100.3960
P3	50.0000	62.3735
Total Generation Cost ($) with PD= 150 MW		1,722.4130
Total Generation Cost ($) with PD= 300 MW		3561.3973

From Table 4.1, the serial number of each generating bus is given in the generator column, the output power of each bus when the power demand is 150MW and 300MW are given in column two and column three respectively. The optimized total of cost of generation for each power demand are given in row five and six respectively. From the Table, it can be observed that the total generation cost for 150 MW is obtained as $1,722.4130. While the total generation cost for 300MW of power demand is obtained to be $3,561.3973. A total power loss of 11.0125W and 4.855196W was obtained when the power demand was 150MW and 300MW respectively.

4.3 Simulation Result of the Proposed Model on IEEE 6-Units ELD Test System

The performance of the monarch butterfly optimization was also evaluated using IEEE 6-unit test system. Just like in the case of 3-unit system, the power demand of this unit was doubled from 700MW to 1400MW. This is to determine the efficiency of developed MBO based ELD system under varying condition of power demand. The subplots showing the cost minimization plots of MBO in the power demand is given in Figure 4.2.

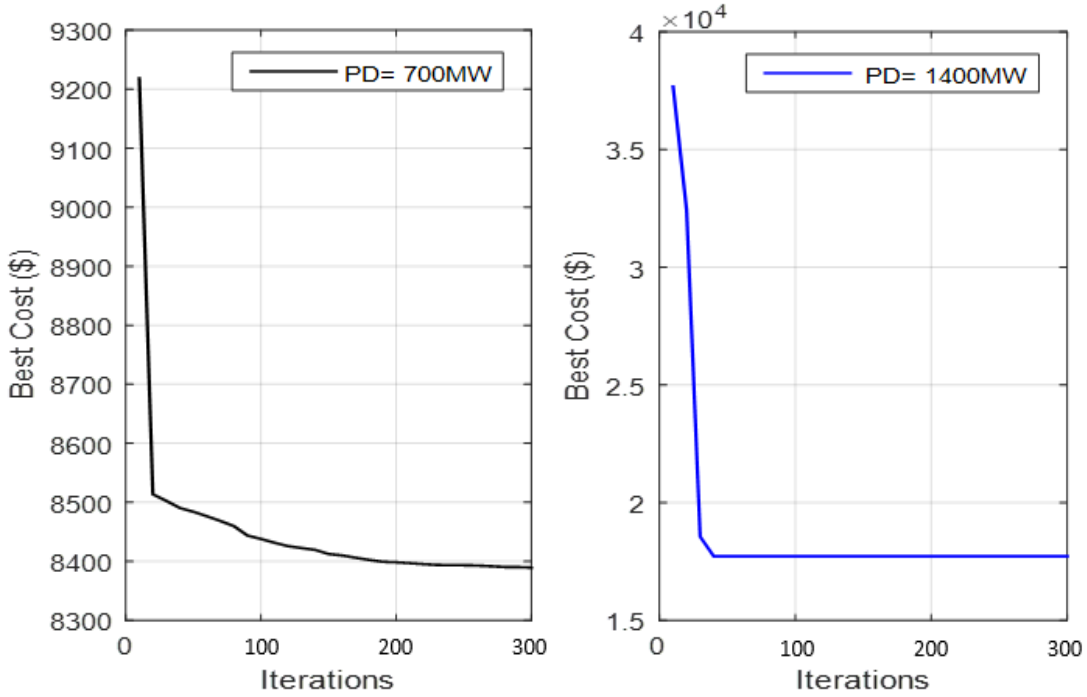


Figure 4.2: Cost Minimization Plot of 6-Unit Test System

The results obtained when the monarch butterfly optimization, was used to optimize the cost of generation on the 6-unit test system is given Figure 4.2. It can be observed that, the MBO optimizes the generation cost for 700 megawatts and 1400 megawatts efficiently. Just like in the case of 3-unit system, the simulation of 6 units was also carried out over a total of 300 iterations. Comparing the subplots response, it can be observed that, the MBO converges at about 200 iterations for the power demand of 700MW and about 50 iterations for power demand of 1400MW respectively. Thereafter, the MBO obtained a constant result all through the optimization process. Detail of the results obtained for the 6-unit system is given in Table 4.2.

Table 4.2: Best Solution Obtained by Proposed MBO Based ELD Algorithm for IEEE 6-unit ELD Test System

Generator	**Power (700MW)**	**Power (1400MW)**
P1	311.0322	499.9998
P2	80.3174	199.9971

P3	156.4824	290.1589
P4	50.0006	149.9999
P5	60.2022	193.8447
P6	50.000	114.8822
Total Generation Cost ($) with PD = 700 (MW)		9978.9427
Total Generation Cost ($) with PD= 1400 (MW)		17720.085

From Table 4.2, it can be observed that the total generation cost for 700 MW is obtained as $9,978.9427. While the total generation cost for 1400MW of power demand is obtained to be $17,720.085. A total of 8.0348W and 48.872W power loss was obtained when the power demand is 700MW and 1400MW respectively.

4.4 Simulation Result of the Proposed Model on IEEE 15-Units ELD Test System

In the third case, the developed monarch butterfly optimization economic load dispatched model was also tested on IEEE 15-unit test system with two different power demand cases. In the first case, a power demand of 2630 MW was selected. While in the second case, a power demand of 5260 MW was selected. The generation cost minimization process for each power demand is shown in Figure 4.3

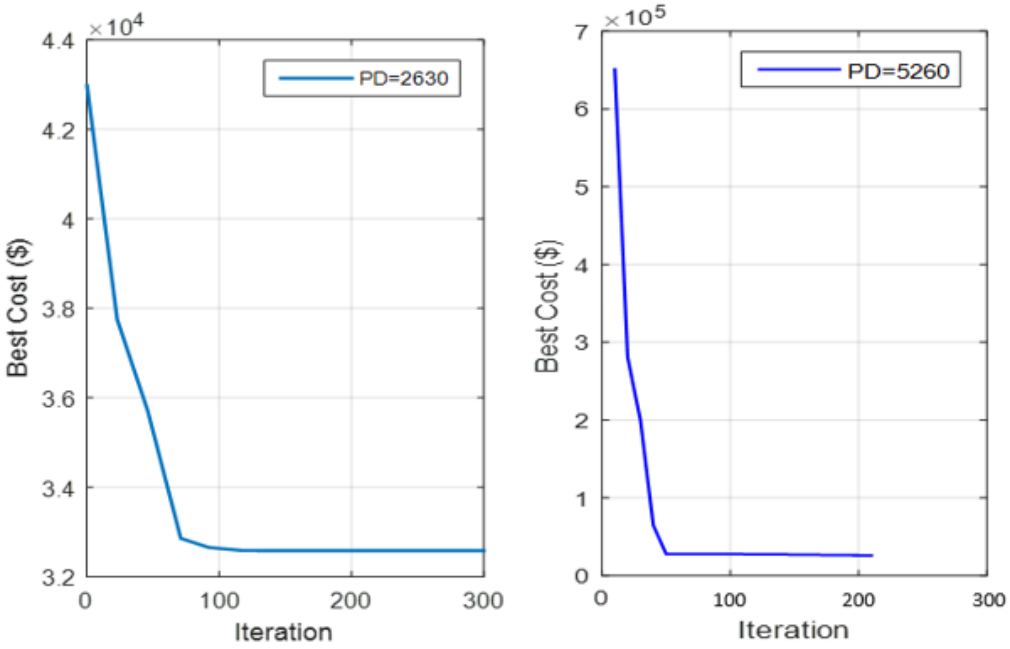


Figure 4.3: Cost Minimization Plot of 15-Units System

From Figure 4.3, it can be observed that the monarch butterfly optimization algorithm minimized the generation cost in both cases of power demand efficiently. When the power demand is 2630 MW, the monarch butterfly optimization algorithm obtained an optimized generation cost at about 200 iterations. Similarly, for power demand of 5260 MW, the optimized generation cost was obtained at around 60 iterations. Detailed results obtained for the 15-unit test system is given in Table 4.3.

Table 4.3: Best Solution Obtained by Proposed MBO Based ELD Algorithm for IEEE 15-Units Test System

Generator	**Power (2630 MW)**	**Power (5260 MW)**
P1	150.0000	671.0000
P2	150.0000	574.0000
P3	84.3698	373.9800
P4	129.9999	374.0000
P5	469.9960	470.0000
P6	459.9983	622.6640
P7	464.9999	544.1127
P8	299.9999	298.7589
P9	161.9999	161.4749
P10	159.9999	430.4070
P11	20.0004	183.9131
P12	20.0000	222.1514
P13	25.0096	223.5380
P14	15.0010	55.0000
P15	15.0000	55.0000
Total Generation Cost ($) with PD= 2630 (MW)		32582.8863
Total Generation Cost ($) with PD=5260 (MW)		22797.1231

In Table 4.3, the power generated by each of the generating unit for 2,630MW power demand and 5,260MW power demand are showed in column two and column three respectively. When the power demand is 2,630MW, the optimized generation obtained was $32,582.8863. Similarly, when the power demand is 5,260 megawatts, the optimized generation cost obtained by the monarch butterfly optimization is $22,797.1231. In the case of 2,630MW, a power loss of 3.6245MW was recorded while in the case of 5,260MW, a total power loss of 0.00106MW was recorded.

4.5 Comparison

In order to verify the performance of the applied model, comparison was done using 15-unit test system. The results obtained was compared with the work of Sayah and Hamouda, (2013) as showed in Table 4.4

Table 4.4: Validation

Generator	Power MBO (MW)	Sayah & Hamouda, (2013) (MW)
P1	150.0000	455.000
P2	150.0000	420.000
P3	84.3698	130.000
P4	129.9999	130.000
P5	469.9960	270.000
P6	459.9983	460.000
P7	464.9999	430.000
P8	299.9999	60.000
P9	161.9999	25.000
P10	159.9999	62.966
P11	20.0004	80.000
P12	20.0000	80.000
P13	25.0096	25.000
P14	15.0010	15.000
P15	15.0000	15.000
Total Generation Cost ($)	32,582.8863	32,588.81

From the validation results shown in Table 4.4, it can be observed that, the generation cost obtained using the MBO was a slightly lower than the cost obtained by the Differential Evolution Particle Swarm Optimization (DEPSO). Furthermore, the DEPSO had a total loss of 27.976MW as compared to 3.6245MW of total losses obtained by the proposed MBO method.

The lower value of loss factor in the case of the applied MBO model on ELD was responsible for the lower cost of generation.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Summary

This dissertation has presented the development of an efficient optimization method based on the intelligent migration behavior of Monarch Butterfly Optimization (MBO) for Economic Load Dispatch (ELD). In the developed model, the initial population of the monarch butterflies was randomly generated. The generated population were divided into two subpopulations residing in land one and land two. The migration and adjustment operators were evaluated and the entire process of MBO was coded in MATLAB R2017a. The economic load dispatch cost generation model was coded as an objective function whose value is to be optimized by the monarch butterfly optimization. The IEEE 3-unist ELD test system, IEEE 6-units ELD test system and IEEE15-units ELD test systems were used to evaluate the performance of the developed model. Simulation was performed over a total of three hundred iterations and the results showed that the monarch optimization algorithm is efficient in determining the generation cost of an economic load dispatch problem.

5.2 Limitation

Some of the limitations encountered during the course of this research are highlighted in the following items

i. The inability to obtain real life generation data to test and evaluate the performance of MBO on the Nigeria power system made it impossible to analyze the cost of generating power using gas resource.

ii. Inability to detect exactly the migration period of the monarch butterflies in both land one and land two also affect the accuracy of the developed model.

iii. The dynamic nature of the load demand which have negative effect on the generation cost was not considered in this study

iv. The non-consideration of generating unit's capacities on the 15- IEEE Test system affects the accuracy of the results as real life ELD dispatches without considering the generators' minimum and maximum capacities impacts on cost function.

5.3 Conclusion

This dissertation has presented an efficient method for generation cost minimization in economic load dispatch problem using monarch butterfly optimization algorithm. The developed model was applied to three standardized IEEE test system which are 3-unit system, 6-unit system and 15-unit test system. Simulation was performed for a total of 300 iterations. Results showed that, the developed MBO obtained an optimized generation cost for each of the test system. Each test systems were simulated for two different power demands. The power demands considered for 3-unit system were 150MW and 300MW; for 6-unit system, the power demand considered were 700MW and 1400MW while the power demand considered for 15-unit system were 2,630MW and 5,260MW respectively. On the 3-unit system, results showed that the proposed MBO obtained a generation cost of $1,722.4130 when the power demand is 150MW and $3,561.3973 when the power demand is 300MW. In the case of 6-unit test system, the MBO obtained a generation cost of $9,978.9427 when the power demand is 700MW while a generation cost of $17,720.085 was obtained for a power demand of 1400MW. Finally, for the 15-unit test system, the power demand of 2630MW and 5260MW were selected and results showed that the MBO obtained a minimized generation cost of $32,582.8863 for 2630MW power demand and $22,797.1231 generation cost for power demand of 5260MW.

The performance of the proposed method was evaluated by comparing the result obtained by MBO with DEPSO results as published in the work of Sayah and Hamouda, (2013) using the

15-unit IEEE test system. This comparison shows for total cost of generation that the proposed MBO model performed better that the DEPSO by 0.0157 (or 0.02%) percent. From these obtained results, it was observed that the developed MBO model for economic load dispatch performed better with increase in the size of the networks which is a sign that the method is promising when used for complex large systems.

5.4 Significant Contribution of the Research Work

The significant contributions of this research work are as follows:

i. The research has developed an MBO-ELD based algorithm for solving Economic Load Dispatch problems to minimized cost.

   The successful implementation of the MBO-ELD based algorithm on IEEE 3, 6 and 15-Units standard test system by considering the improvement based on different demand profile of low dimensional generating units of 150MW, 700MW, 1400MW and 2630MW respectively. The results obtained from the proposed model showed a significant reduction by 0.0157MW equivalent to 0.02% in total cost of generation when compared with the work of Sayah and Hamouda, (2013).

5.5 Recommendations for Further Work

Although this dissertation has presented the application of monarch butterfly optimization algorithm to economic load dispatch model, the following areas are recommended for consideration in future work

1) The performance of monarch butterfly optimization algorithm on solving economic load dispatch problems can further be verified on larger test system with dimensionality greater than 15.

2) The monarch butterfly optimization algorithm can be applied to other type problems such as load frequency control, distributed generation etc.

REFERENCES

Akay B., and Dervis K., (2012). A modified artificial bee colony algorithm for real-parameter optimization. Information Sciences 192:120-142.

Alam, M. N., K. Kumar, and A. Mathur (2015). Economic load dispatch considering valve-point effects using time varying constriction factor-based particle swarm optimization. 2015 IEEE UP Section Conference on Electrical Computer and Electronics (UPCON), 2015, pp. 1-6.

Alam, M. N., A. Mathur, and K. Kumar (2016). Economic load dispatch using a differential particle swarm optimization. 2016 *IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES),* 2016, pp. 1-5.

Babaoˇglu, I., Fındık, O., Ülker, E., & Aygül, N. (2012). A novel hybrid classification method with particle swarm optimization and k-nearest neighbor algorithm for diagnosis of coronary artery disease using exercise stress test data. *International Journal of Innovative Computing, Information and Control*, *8*(5), 3467-3475.

Bahrani, L. T. A., and J. C. Patra (2015). Orthogonal PSO Algorithm for Economic Dispatch of Power under Power Grid Constraints. 2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015, pp. 14-19.

Chen, X., and Y. Li (2007). A modified PSO structure resulting in high exploration ability with convergence guaranteed. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics 37(5):1271-1289.

Das, B., and Sengupta T. K., (2015). Economic load dispatch using PSO and TLBO. Michael Faraday IET International Summit 2015, 2015, pp. 212-219.

Dasgupta, K., S. Banerjee, and C. K. Chanda (2015). An analysis of economic load dispatch with prohibited zone constraint using different algorithms. Michael Faraday IET International Summit 2015, 2015, pp. 193-197.

Dasgupta, K., Banerjee, S., & Chanda, C. K. (2016, January). Economic load dispatch with prohibited zone and ramp-rate limit constraints—A comparative study. In *Control, Measurement and Instrumentation (CMI), 2016 IEEE First International Conference on* (pp. 26-30). IEEE.

Eberhart R., and James K., (1995). A new optimizer using particle swarm theory. Micro Machine and Human Science, 1995. MHS'95., IEEE Proceedings of the Sixth International Symposium on Micromachine and Human Science, 1995, pp. 39-43.

Elyas, S. H., Mandal, P., Haque, A. U., Giani, A., & Tseng, T. L. B. (2014, September). A new hybrid optimization algorithm for solving economic load dispatch problem with valve-point effect. In *North American Power Symposium (NAPS), 2014* (pp. 1-6). IEEE.

Gautham S., and J. Rajamohan (2016). Economic Load Dispatch using Novel Bat Algorithm. 2016 IEEE 1st International Conference on Power Electronics, Intelligent Control and Energy Systems (ICPEICES), 2016, pp. 1-4.

Ghasemi, H. S. A. (2012). Application of MOPSO for economic load dispatch solution with transmission losses. *Int J Eng SciTech*, *4*, 27-34.

Hosseinnezhad V., and Ebrahim B., (2013). Economic load dispatch using θ-PSO. International Journal of Electrical Power & Energy Systems 49:160-169.

Jabr R. A., Alun H. C., and Brian J. C., (2000). A homogeneous linear programming algorithm for the security constrained economic dispatch problem. Power Systems, IEEE Transactions on 15(3):930-936.

Kamboj V. K., Bath S.K., and Dhillon J.S., (2016). Solution of non-convex economic load dispatch problem using Grey Wolf Optimizer. Neural Computing and Applications 27(5):1301-1316.

Karaboga D., (2005). An idea based on honey bee swarm for numerical optimization. Technical report-tr06, Erciyes University, Engineering Faculty, Computer Engineering Department.

Karaboga, Dervis, and Bahriye Basturk (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. Journal of global optimization 39(3):459-471.

Kassim Ali Al-Anbarri, and Saeed Majeed Wafaa (2012). A Reliable Load Flow Method for Radial Distribution System. Journal of Engineering and Development 16(2):248-265.

Kumar A. N., and Suchitra D., (2011). AI based Economic Load Dispatch incorporating wind power penetration. Proceedings of the 2011 International Conference on Electrical Engineering and Informatics, 2011, pp. 1-8.

Kumar N., Nangia U., and Sahay K. B., (2014). Economic load dispatch using improved particle swarm optimization algorithms. 2014 6th IEEE Power India International Conference (PIICON), 2014, pp. 1-6.

Mahdi, F. P., Vasant, P., Rahman, M. M., Abdullah-Al-Wadud, M., Watada, J., & Kallimani, V. (2017, February). Quantum particle swarm optimization for multiobjective combined economic emission dispatch problem using cubic criterion function. In *Imaging, Vision & Pattern Recognition (icIVPR), 2017 IEEE International Conference on* (pp. 1-5). IEEE.

Nawaz, A., Saleem, N., Mustafa, E., & Khan, U. A. (2017). An efficient global technique for solving the network constrained static and dynamic economic dispatch problem. *Turkish Journal of Electrical Engineering & Computer Sciences*, *25*(1), 73-82.

Niu, H., Dai, Y., & Peng, X. (2012). Particle swarm optimization with adaptive parameters and boundary constraints. *IJ Engineering and Manufacturing*, *4*, 19-28.

Park, J. H., Kim, Y. S., Eom, I. K., & Lee, K. Y. (1993). Economic load dispatch for piecewise quadratic cost function using Hopfield neural network. *IEEE transactions on power systems*, *8*(3), 1030-1038.

Rani, C., & Kothari, D. P. (2012, December). Dynamic Economic Emission Dispatch problem with valve-point effect. In *Emerging Trends in Electrical Engineering and Energy Management (ICETEEEM), 2012 International Conference on* (pp. 109-114). IEEE.

Rizwana, J., Jeevitha, R., Venkatesh, R., & Parthiban, K. S. (2015, December). Minimization of fuel cost in solving the power economic dispatch problem including transmission losses by using modified Particle Swarm Optimization. In *Computational Intelligence and Computing Research (ICCIC), 2015 IEEE International Conference on* (pp. 1-4). IEEE.

Saber, A. Y., & Rahman, D. M. F. (2011, July). Economic load dispatch using particle swarm differential evolution optimization. In *Power and Energy Society General Meeting, 2011 IEEE* (pp. 1-8). IEEE.

Santra, D., Mukherjee, A., Sarker, K., & Chatterjee, D. (2016, October). Hybrid PSO-ACO algorithm to solve economic load dispatch problem with transmission loss for small scale power system. In *Intelligent Control Power and Instrumentation (ICICPI), International Conference on* (pp. 226-230). IEEE.

Sayah S. & hamouda A. (2013). A hybrid differential evolution based on particle swarm optimization for nonconvex economic load dispatch problems. *In Journal of Applied Soft Computing* 13 (2013)1608-1619.

Shahir, F. M., Farsadi, M., Zafari, H., & Sadighmanesh, A. (2015, November). Solving economic emission load dispatch problems using particle swarm optimization with smart inertia factor. In *Electrical and Electronics Engineering (ELECO), 2015 9th International Conference on* (pp. 500-504). IEEE.

Sharma, U., & Moses, B. (2016, March). Analysis and optimization of economic load dispatch using soft computing techniques. In *Electrical, Electronics, and Optimization Techniques (ICEEOT), International Conference on* (pp. 4035-4040). IEEE.

Singh, N., & Kumar, Y. (2013, February). Economic load dispatch with environmental emission using MRPSO. In *Advance Computing Conference (IACC), 2013 IEEE 3rd International* (pp. 995-999). IEEE.

Sinha, N., Chakrabarti, R., & Chattopadhyay, P. K. (2003). Evolutionary programming techniques for economic load dispatch. *IEEE Transactions on evolutionary computation*, *7*(1), 83-94.

Subathra, M. S. P., Selvan, S. E., Victoire, T. A. A., Christinal, A. H., & Amato, U. (2015). A hybrid with cross-entropy method and sequential quadratic programming to solve economic load dispatch problem. *IEEE Systems Journal*, *9*(3), 1031-1044.

Suman, M., Rao, M. V. G., Hanumaiah, A., & Rajesh, K. (2016). Solution of economic load dispatch problem in power system using Lambda Iteration and back propagation neural network methods. *International Journal on Electrical Engineering and Informatics*, *8*(2), 347.

Tiwari, A., & Pandit, M. (2016, March). Bid based economic load dispatch using symbiotic organisms search algorithm. In *Engineering and Technology (ICETECH), 2016 IEEE International Conference on* (pp. 1073-1078). IEEE.

Trivedi, I. N., Bhoye, M., Bhesdadiya, R. H., Jangir, P., Jangir, N., & Kumar, A. (2016a). An emission constraint environment dispatch problem solution with microgrid using whale optimization algorithm. In *Power Systems Conference (NPSC), 2016 National* (pp. 1-6). IEEE.

Trivedi, I. N., Purohit, S. N., Jangir, P., & Bhoye, M. T. (2016b). Environment dispatch of distributed energy resources in a microgrid using JAYA algorithm. In *Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB), 2016 2nd International Conference on* (pp. 224-228). IEEE.

Wang, G. G., Hao, G. S., Cheng, S., & Qin, Q. (2016). A discrete monarch butterfly optimization for Chinese TSP problem. In *International Conference in Swarm Intelligence* (pp. 165-173). Springer, Cham.

Wang, G. G., Zhao, X., & Deb, S. (2015, November). A novel monarch butterfly optimization with greedy strategy and self-adaptive. In *Soft Computing and Machine Intelligence (ISCMI), 2015 Second International Conference on* (pp. 45-50). IEEE.

Wang, L., & Li, L. P. (2013). An effective differential harmony search algorithm for the solving non-convex economic load dispatch problems. *International Journal of Electrical Power & Energy Systems*, *44*(1), 832-843.

Wood, A. J., & Wollenberg, B. F. (2015). 'Power Generation, Operation and Control', John Wiley & Sons, Inc., 2003.

Zhang, Y., Gong, D. W., & Ding, Z. (2012). A bare-bones multi-objective particle swarm optimization algorithm for environmental/economic dispatch. *Information sciences*, *192*, 213-227.

Appendix A: MATLAB CODE OF MBO

```
function [MinCost] = MBO(ProblemFunction, DisplayFlag, RandSeed)

% Monarch Butterfly Optimization (MBO) software for minimizing a general
function
% The fixed generation is considered as termination condition.

% INPUTS: ProblemFunction is the handle of the function that returns
%         the handles of the initialization, cost, and feasibility
functions.
%         DisplayFlag = true or false, whether or not to display and plot
results.
%         ProbFlag = true or false, whether or not to use probabilities to
update emigration rates.
%         RandSeed = random number seed
% OUTPUTS: MinCost = array of best solution, one element for each generation
%          Hamming = final Hamming distance between solutions
% CAVEAT: The "ClearDups" function that is called below replaces duplicates
with randomly-generated
%         individuals, but it does not then recalculate the cost of the
replaced individuals.

tic

if ~exist('ProblemFunction', 'var')
    ProblemFunction = @Ackley;
end
if ~exist('DisplayFlag', 'var')
    DisplayFlag = true;
end
if ~exist('RandSeed', 'var')
    RandSeed = round(sum(100*clock));
end

[OPTIONS, MinCost, AvgCost, InitFunction, CostFunction, FeasibleFunction,
...
    MaxParValue, MinParValue, Population] = Init(DisplayFlag,
ProblemFunction, RandSeed);
n=input('ENTER 1 TO RUN MBO FOR ELD PROBLEM:=');
% % % % % % % % % % % %             Initial parameter setting          % % %
% % % % % % % % %%%%
%% Initial parameter setting
Keep = 2; % elitism parameter: how many of the best habitats to keep from
one generation to the next
maxStepSize = 1.0;        %Max Step size
partition = OPTIONS.partition;
numButterfly1 = ceil(partition*OPTIONS.popsize);  % NP1 in paper
numButterfly2 = OPTIONS.popsize - numButterfly1; % NP2 in paper
period = 1.2; % 12 months in a year
Land1 = zeros(numButterfly1, OPTIONS.numVar);
Land2 = zeros(numButterfly2, OPTIONS.numVar);
BAR = partition; % you can change the BAR value in order to get much better
performance
% % % % % % % % % % %       End of Initial parameter setting       % % % %
% % % % % % % %%
%%
```

```
% % % % % % % % % % %             Begin the optimization loop        % % %
% % % % % %%%%
% Begin the optimization loop
if n==1
    P1
end
if n==0
for GenIndex = 1 : OPTIONS.Maxgen

    % % % % % % % % % % %            Elitism Strategy           % % % % %
% % % % % %%%%%
    %% Save the best monarch butterflis in a temporary array.
    for j = 1 : Keep
        chromKeep(j,:) = Population(j).chrom;
        costKeep(j) = Population(j).cost;
    end
    % % % % % % % % % % %       End of  Elitism Strategy      % % % % % %
% % % % % %%%%
    %%

    % % % % % % % % % % %    Divide the whole population into two
subpopulations % % % %%%
    %% Divide the whole population into Population1 (Land1) and Population2
(Land2)
    % according to their fitness.
    % The monarch butterflies in Population1 are better than or equal to
Population2.
    % Of course, we can randomly divide the whole population into
Population1 and Population2.
    % We do not test the different performance between two ways.
    for popindex = 1 : OPTIONS.popsize
        if popindex <= numButterfly1
            Population1(popindex).chrom = Population(popindex).chrom;
        else
            Population2(popindex-numButterfly1).chrom =
Population(popindex).chrom;
        end
    end
    % % % % % % % % % %    End of Divide the whole population into two
subpopulations  % % %%%
    %%

    % % % % % % % % % % %%            Migration operator          % % % %
% % % % % % %%%%
    %% Migration operator
    for k1 = 1 : numButterfly1
        for parnum1 = 1 : OPTIONS.numVar
            r1 = rand*period;
            if r1 <= partition
                r2 = round(numButterfly1 * rand + 0.5);
                Land1(k1,parnum1) = Population1(r2).chrom(parnum1);
            else
                r3 = round(numButterfly2 * rand + 0.5);
                Land1(k1,parnum1) = Population2(r3).chrom(parnum1);
            end
        end %% for parnum1
        NewPopulation1(k1).chrom =  Land1(k1,:);
    end  %% for k1
```

```
    % % % % % % % % % % %%%       End of Migration operator      % % % % %
% % % % % %%%
    %%

    % % % % % % % % % % %             Evaluate NewPopulation1          % %
% % % % % % % % % %%
    %% Evaluate NewPopulation1
    SavePopSize = OPTIONS.popsize;
    OPTIONS.popsize = numButterfly1;
    % Make sure each individual is legal.
    NewPopulation1 = FeasibleFunction(OPTIONS, NewPopulation1);
    % Calculate cost
    NewPopulation1 = CostFunction(OPTIONS, NewPopulation1);
    OPTIONS.popsize = SavePopSize;
    % % % % % % % % % % %       End of Evaluate NewPopulation1      % % %
% % % % % % % % %%
    %%

    % % % % % % % % % % %             Butterfly adjusting operator
% % % % % % % % % % %%
    %% Butterfly adjusting operator
    for k2 = 1 : numButterfly2
        scale = maxStepSize/(GenIndex^2); %Smaller step for local walk
        StepSzie = ceil(exprnd(2*OPTIONS.Maxgen,1,1));
        delataX = LevyFlight(StepSzie,OPTIONS.numVar);
        for parnum2 = 1:OPTIONS.numVar,
            if (rand >= partition)
                Land2(k2,parnum2) = Population(1).chrom(parnum2);
            else
                r4 = round(numButterfly2*rand + 0.5);
                Land2(k2,parnum2) =  Population2(r4).chrom(1);
                if (rand > BAR) % Butterfly-Adjusting rate
                    Land2(k2,parnum2) =  Land2(k2,parnum2) +
scale*(delataX(parnum2)-0.5);
                end
            end
        end  %% for parnum2
        NewPopulation2(k2).chrom =  Land2(k2,:);
    end %% for k2
    % % % % % % % % % % %       End of Butterfly adjusting operator      %
% % % % % % % % % %
    %%

    % % % % % % % % % % %             Evaluate NewPopulation2          % %
% % % % % % % % % %%
    %% Evaluate NewPopulation2
    SavePopSize = OPTIONS.popsize;
    OPTIONS.popsize = numButterfly2;
    % Make sure each individual is legal.
    NewPopulation2 = FeasibleFunction(OPTIONS, NewPopulation2);
    % Calculate cost
    NewPopulation2 = CostFunction(OPTIONS, NewPopulation2);
    OPTIONS.popsize = SavePopSize;
    % % % % % % % % % % %       End of Evaluate NewPopulation2      % % %
% % % % % % % % %%
    %%

    % % % % % %  Combine two subpopulations into one and rank monarch
butterflis       % % % % % %
```

```
    %% Combine Population1 with Population2 to generate a new Population
    Population = CombinePopulation(OPTIONS, NewPopulation1, NewPopulation2);
    % Sort from best to worst
    Population = PopSort(Population);
    % % % % % %     End of Combine two subpopulations into one and rank
monarch butterflis  % %% % %
    %%

    % % % % % % % % % % %            Elitism Strategy          % % % % % %
% % % % %%% %% %
    %% Replace the worst with the previous generation's elites.
    n = length(Population);
    for k3 = 1 : Keep
        Population(n-k3+1).chrom = chromKeep(k3,:);
        Population(n-k3+1).cost = costKeep(k3);
    end % end for k3
    % % % % % % % % % % %     End of  Elitism Strategy      % % % % % % %
% % % % %%% %% %
    %%
    % % % % % % % % %           Precess and output the results          %
% % % % % % % % % %%%
    % Sort from best to worst
    Population = PopSort(Population);
    % Compute the average cost
    [AverageCost, nLegal] = ComputeAveCost(Population);
    % Display info to screen
    MinCost = [MinCost Population(1).cost];
    AvgCost = [AvgCost AverageCost];
    if DisplayFlag
        disp(['The best and mean of Generation # ', num2str(GenIndex), ' are
',...
            num2str(MinCost(end)), ' and ', num2str(AvgCost(end))]);
    end
    % % % % % % % % % %    End of Precess and output the results
%%%%%%%%% %% %
    %%
end % end for GenIndex
Conclude1(DisplayFlag, OPTIONS, Population, nLegal, MinCost, AvgCost);
end
toc
% % % % % % % % %     End of Monarch Butterfly Optimization implementation
%%%% %% %
%%

function [delataX] = LevyFlight(StepSize, Dim)

%Allocate matrix for solutions
delataX = zeros(1,Dim);

%Loop over each dimension
for i=1:Dim
    % Cauchy distribution
    fx = tan(pi * rand(1,StepSize));
    delataX(i) = sum(fx);
end
```

Appendix B: MATLAB CODE OF MBO PARAMETER INITIALIZATION

```
function [OPTIONS, MinCost, AvgCost, InitFunction, CostFunction,
FeasibleFunction, ...
    MaxParValue, MinParValue, Population] = Init(DisplayFlag,
ProblemFunction, RandSeed)

% Initialize population-based optimization software.

% WARNING: some of the optimization routines will not work if population
size is odd.
OPTIONS.popsize = 50; % total population size
OPTIONS.Maxgen = 1000; % generation count limit
OPTIONS.numVar =  20; % number of vriables in each population member
OPTIONS.MaxFEs = 1E4; % number of Function Evaluations (FEs)
OPTIONS.partition = 5/12;  % the percentage of population for MBO

if ~exist('RandSeed', 'var')
    RandSeed = round(sum(100*clock));
end
rand('state', RandSeed); % initialize random number generator
if DisplayFlag
    disp(['random # seed = ', num2str(RandSeed)]);
end

% Get the addresses of the initialization, cost, and feasibility functions.
[InitFunction, CostFunction, FeasibleFunction] = ProblemFunction();
% Initialize the population.
[MaxParValue, MinParValue, Population, OPTIONS] = InitFunction(OPTIONS);
% Make sure the population does not have duplicates.
Population = ClearDups(Population, MaxParValue, MinParValue);
% Compute cost of each individual
Population = CostFunction(OPTIONS, Population);
% Sort the population from most fit to least fit
Population = PopSort(Population);
% Compute the average cost
AverageCost = ComputeAveCost(Population);
% Display info to screen
MinCost = [Population(1).cost];
AvgCost = [AverageCost];
if DisplayFlag
    disp(['The best and mean of Generation # 0 are ', num2str(MinCost(end)),
' and ', num2str(AvgCost(end))]);
end

return;
```

Appendix C: MATLAB CODE FOR SORTING POPULATION

```
function [Population, indices] = PopSort(Population)
% Sort the population members from best to worst
popsize = length(Population);
Cost = zeros(1, popsize);
indices = zeros(1, popsize);
for i = 1 : popsize
    Cost(i) = Population(i).cost;
end
[Cost, indices] = sort(Cost, 2, 'ascend');
Chroms = zeros(popsize, length(Population(1).chrom));
for i = 1 : popsize
    Chroms(i, :) = Population(indices(i)).chrom;
end
for i = 1 : popsize
    Population(i).chrom = Chroms(i, :);
    Population(i).cost = Cost(i);
end
```

Appendix D: MATLAB CODE OF ELD COST FUNCTION

```
% This program solves the economic dispatch with Bmn coefficients MBO

  function[ F P1 Pl]=eldcost(x)
global objfun D ub lb data B  Pd
n=length(data(:,1));
[m n1]=size(x);
P=x(1:m,2:n);
B11=B(1,1);
B1n=B(1,2:n);
Bnn=B(2:n,2:n);
A=B11;
BB1=2*B1n*P';
B1=(BB1-1)';
C1=(P*Bnn*P');
C1=diag(C1);
C=Pd-(sum(P'))'+C1;
 A=A*ones(m,1);
for i=1:m
    y=[A(i) B1(i) C(i)];
x1(i,:)=roots(y);
 x2(i)=(abs(min(x1(i,:))))';
 if x2(i)>data(1,5)
     x2(i)=data(1,5);
 else
 end
   if x2(i)<data(1,4)
x2(i)=data(1,4);
   else
   end
end
 P1=[x2' P];
 a1=data(:,1);
 b1=data(:,2);
 c1=sum(data(:,3));
 F=P1.*P1*a1+P1*b1+c1;
 Pl1=(P1*B*P1').';
 Pl=diag(Pl1);
 lam=abs(sum(P1')'-Pd-Pl);
 F=(F)+1000*lam;
```